#### Project report on

**Web Vulnerability Scanner (Security Analysis)**

**A Dissertation submitted in partial fulfillment of the Academic requirements for the award of the degree of**

**Bachelor of Technology**

## In

**Computer Science & Engineering (Cyber Security)**

**Submitted by**

Ch. Venkat Ramana (22H51A6211)

Ch. Vivekananda (22H51A6216)

G. Pooja (22H51A6217)

**Under the esteemed Guidance of**

**Dr.R.Venkateswara Reddy (Associate Professor and HOD,CSC)**

**Department of Cyber Security**

**CMR COLLEGE OF ENGINEERING & TECHNOLOGY**

**(Autonomous)**

**(NAAC Accredited with ‘A+’ Grade & NBA Accredited) (Approved by AICTE, Permanently Affiliated to JNTU Hyderabad)**

**KANDLAKOYA, MEDCHAL ROAD, HYDERABAD-501401**

#### CMR COLLEGE OF ENGINEERING & TECHNOLOGY

**(Autonomous)**

**(NAAC Accredited with ‘A+’ Grade & NBA Accredited) (Approved by AICTE, Permanently Affiliated to JNTU Hyderabad)**

**KANDLAKOYA, MEDCHAL ROAD, HYDERABAD-501401**

**DEPARTMENT OF CYBER SECURITY**



#### CERTIFICATE

This is to certify that the Mini Project -1 report entitled “**Web Vulnerability Scanner (Security Analysis)**” being submitted by **Ch. Venkat Ramana(22H51A6211), Ch. Vivekananda (22H51A6216), G. Pooja(22H51A6217)** in partial fulfillment for the award of **Bachelor of Technology in Computer Science and Engineering (Cyber Security)** is a record of bonafide work carried out his/her under my guidanceand supervision.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree.

K. Sharath Kumar Dr. R. Venkateswara Reddy

Assistant Professor Associate Professor & HOD

Dept. of CSC Dept. of CSC

#### ACKNOWLEDGEMENT

With great pleasure I want to take this opportunity to express my heartfelt gratitude to all the people

`who helped in making this project a grand success.

I am grateful to **K. Sharath Kumar**, Assistant Professor, Dept. of Computer Science and Engineering for her valuable suggestions and guidance during the execution of this project.

I would like to thank **Dr. R. Venkateswara Reddy**, Head of the Department of Computer Science and Engineering, for his moral support throughout the period of my study in CMRCET.

I am highly indebted to **Major Dr. V.A. NARAYANA**, Principal CMRCET, for giving permission to carry out this project in a successful and fruitful way.

I would like to thank the Teaching & Non- teaching staff of the Department of Computer Science and Engineering for their co-operation.

Finally, I express my sincere thanks to **Mr. CH. GOPAL REDDY**, Secretary, CMR Group of Institutions, for his continuous care. I sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project work.

Ch. Venkat Ramana

(22H51A6211)

Ch. Vivekananda

(22H51A6216)

G. Pooja

(22H51A6217)

#### ABSTRACT

* Web applications are increasingly becoming targets for cyber-attacks, with many attacks SQL injection being one of the most prevalent and damaging vulnerabilities.
* SQL injection attacks allow malicious actors to interfere with the queries that an application makes to its database, potentially leading to unauthorized access, data leaks, and even complete system compromise.
* This detailed abstract outlines the development and implementation of a Web Vulnerability Scanner specifically designed to detect SQL injection vulnerabilities in web applications..
* The Web Vulnerability Scanner will utilize advanced techniques to probe web applications for SQL injection points. By generating a wide array of crafted SQL payloads, the scanner will test various injection methods including Union-based, Error-based, and Blind SQL injection.
* The scanner's sophisticated analysis engine will meticulously examine server responses to identify anomalies indicative of SQL injection vulnerabilities, ensuring comprehensive detection across different application layers and query structures.

**Table Of Content**

|  |  |  |
| --- | --- | --- |
| **CHAPTERS** | **DESCRIPTION** | **PAGE NUMBERS** |
| 1 | INTRODUCTION | 2 |
| 1.1 | AIM | 3 |
| 1.2 | SCOPE | 4 |
| 2 | LITERATURE REVIEW | 6 |
| 3 | EXISTING SOLUTIONS | 8 |
| 4 | PROPOSED SYSTEM | 11 |
| 4.1 | REQUIREMENT ANALYSIS | 12 |
| 4.1.1 | HARDWARE REQUIREMENTS | 12 |
| 4.1.2 | SOFTWARE REQUIREMENTS | 12 |
| 4.2 | MERITS AND DEMERITS | 13 |
| 5 | DESIGN DESCRIPTION | 15 |
| 5.1 | CONCEPTUAL DESIGN | 15 |
| 6 | IMPLEMENTATION AND DISCUSSION | 16 |
| 6.1 | IMPLEMENTATION | 17 |
| 7 | RESULT | 20 |
| 8 | CONCLUSION AND FUTURE ENHANCEMENT | 21 |
| 8.1 | CONCLUSION | 21 |
| 8.2 | ENHANCEMENT | 21 |
| 8.3 | REFERENCES | 24 |

# CHAPTER 1

1

#### INTRODUCTION

* The rapid proliferation of web applications has brought about significant advancements in how businesses and individuals interact with digital platforms.
* However, this growth has also led to an increased exposure to various cyber threats, with SQL injection attacks being one of the most common and dangerous.
* SQL injection is a type of attack where malicious actors manipulate SQL queries made to a database by injecting malicious code.
* This can lead to unauthorized access, data leaks, and, in severe cases, full control over the web application's backend database.
* SQL injection and many more attacks exploit vulnerabilities in the input fields of web applications, such as login forms, search boxes, and URL parameters.
* By crafting malicious input that alters the intended SQL query, attackers can bypass authentication mechanisms, retrieve sensitive information, modify database records, and execute administrative operations.
* The impact of such attacks can be devastating, resulting in significant financial losses, reputational damage, and legal liabilities for affected organizations.

#### AIM

* The primary aim of this project is to develop a robust and efficient Web Vulnerability Scanner specifically designed to detect SQL injection, vulnerabilities in web applications.
* The scanner will serve as a critical tool in enhancing web application security by automating the detection process and providing comprehensive, actionable insights into potential vulnerabilities.
* sophisticated detection algorithms and response analysis techniques to minimize false positives and ensure accurate identification of genuine vulnerabilities.
* Design the scanner to operate efficiently, with optimal speed and resource usage, ensuring it can be deployed effectively in various environments without causing significant performance overhead.
* Ensure that the scanner operates safely, avoiding any disruptive actions that could negatively impact the normal functioning of the web application during the scanning process.

#### SCOPE

* The scope of this project encompasses the design, development, and deployment of a Web Vulnerability Scanner specifically focused on detecting SQL injection attack, vulnerabilities in web applications. The project aims to provide a comprehensive solution that addresses various aspects of SQL injection, from identifying potential entry points to offering detailed remediation guidance. The scope includes the following key components:

#### ****Testing and Validation****

* **Controlled Environment Testing**: Validate the scanner's effectiveness by testing it against intentionally vulnerable web applications to ensure it accurately detects SQL injection vulnerabilities.
* **Performance Testing**: Assess the scanner's performance in terms of speed, resource consumption, and scalability to ensure it operates efficiently in various environments.
* **False Positive Minimization**: Implement measures to minimize false positives, ensuring that the scanner accurately identifies genuine vulnerabilities.

# CHAPTER 2

#### LITERATURE REVIEW

* The literature review provides an overview of existing research and developments in the field of web application security, specifically focusing on SQL injection vulnerabilities and the tools designed to detect and mitigate these threats. This section examines various studies, methodologies, and technologies that have been explored to understand the current landscape and identify gaps that the proposed Web Vulnerability Scanner aims to address.
* **Mechanics of SQL Injection:** Researchers have extensively documented how SQL injection attacks exploit vulnerabilities in the way web applications construct SQL queries. These attacks can bypass authentication, retrieve sensitive data, and execute arbitrary commands on the database.
* **Types of SQL Injection:** Studies categorize SQL injection into several types, including Union-based, Error-based, and Blind SQL injection, each with unique characteristics and detection challenges.
* **Source Code Analysis:** Static analysis tools inspect the source code of web applications to identify potential SQL injection vulnerabilities. These tools rely on predefined patterns and rules to detect insecure coding practices.
* **Limitations:** While effective in identifying vulnerabilities in the source code, static analysis tools often produce false positives and may not detect runtime issues.

# CHAPTER 3

#### EXISTING SOLUTION

#### 1. ****SQLMap****

* SQLMap is an open-source tool that automates the process of detecting and exploiting SQL injection vulnerabilities in web applications. It is one of the most popular and comprehensive tools available for SQL injection testing.

.



**Fig 1:** SQL MAP

#### ****Burp Suite****

* Burp Suite is a comprehensive web vulnerability scanner that includes various tools for security testing, including a module for detecting SQL injection vulnerabilities.



**Fig 2:** BURPSUITE

#### 3. ****Wapiti****

* Wapiti is an open-source tool that performs black-box testing to detect various types of vulnerabilities, including SQL injection.



**Fig 3:** WAPITI

#### ****4 .Acunetix****

* Acunetix is a commercial web vulnerability scanner that offers comprehensive scanning capabilities, including the detection of SQL injection vulnerabilities.



##### 

**Fig 4:** ACUNETIX

# CHAPTER 4

#### PROPOSED SYSTEM

Nikto is a renowned open-source web server scanner utilized extensively to uncover potential vulnerabilities within web servers and applications. It specializes in detecting issues such as outdated software versions, common misconfigurations, and known vulnerabilities that could compromise the security of web environments. Installation is typically straightforward through standard package managers like apt or yum on Linux distributions. Once installed, Nikto operates through command-line interfaces, with a primary command structure of nikto -h <target\_host> where <target\_host> denotes the URL or IP address of the web server under scrutiny.

The tool offers a variety of options to enhance its functionality and scope. For instance, -o allows users to specify output formats such as HTML, XML, or CSV to store scan results in a structured manner. It supports scanning specific ports using -p <port> options, making it versatile for servers operating on non-default ports or multiple services. Security protocols can be managed with options like -ssl to enforce SSL/TLS connections or -no\_ssl to bypass them entirely. Authentication mechanisms are supported through -id <id:pass> for basic authentication and -mutate for assessing credential mutation scenarios.

Nikto's scan results are comprehensive, providing detailed reports that outline discovered vulnerabilities, their severity levels, and recommended actions for remediation. Each finding is accompanied by actionable insights, making it valuable for both security professionals and system administrators in addressing potential security risks effectively. It's crucial, however, to interpret results carefully, as automated scanners like Nikto can occasionally generate false positives that require verification. Regular updates are essential to ensure Nikto includes the latest vulnerability checks and improvements, aligning with evolving web security standards and threat landscapes.

#### REQUIREMENT ANALYSIS

##### Software Requirements

* + - * Windows 7 or later, Linux, or macOS
      * Web application framework
      * DVWA login
      * Scanning Tools

##### Hardware Requirements

* + - * System 32 or 64 bit with 4 GB or 8 GB RAM
      * Network Security Devices
      * RAM





**Fig 5:** Kali Linux **Fig 6:** DVWA

##### MERITS AND DEMERITS Merits:

* + - * Comprehensive Vulnerability Detection
      * Open-Source and Free
      * Customizable Scanning Options
      * Platform Compatibility
      * Community Collaboration

##### Demerits:

* + - * Potential False Positives
* Limited Web Application Testing
* Dependency on Network Connectivity

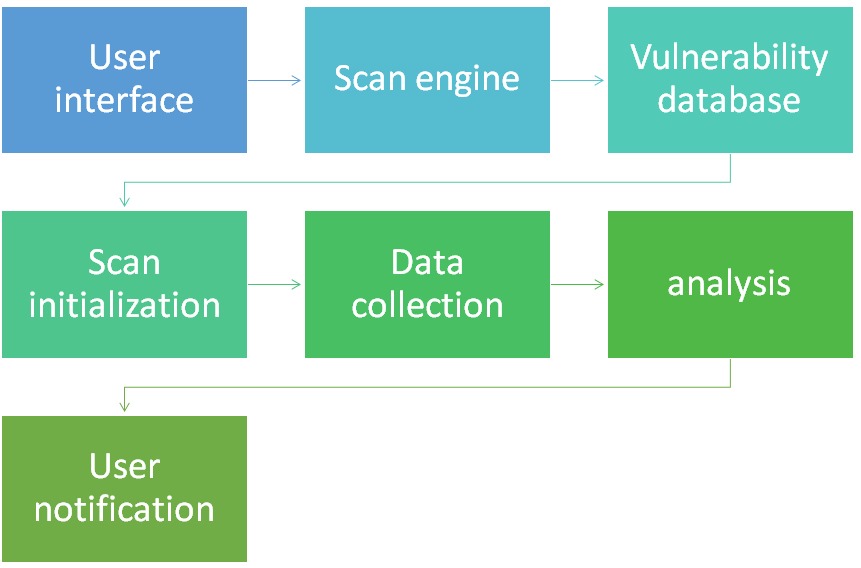
# CHAPTER 5

14

#### DESIGN DESCRIPTION

##### 5.1 CONCEPTUAL DESIGN

The diagram shows the steps involved in Web Vulnerability Scanner (Security Analysis)



**Fig 7:** Steps for Web Vulnerability Scanner

# CHAPTER 6

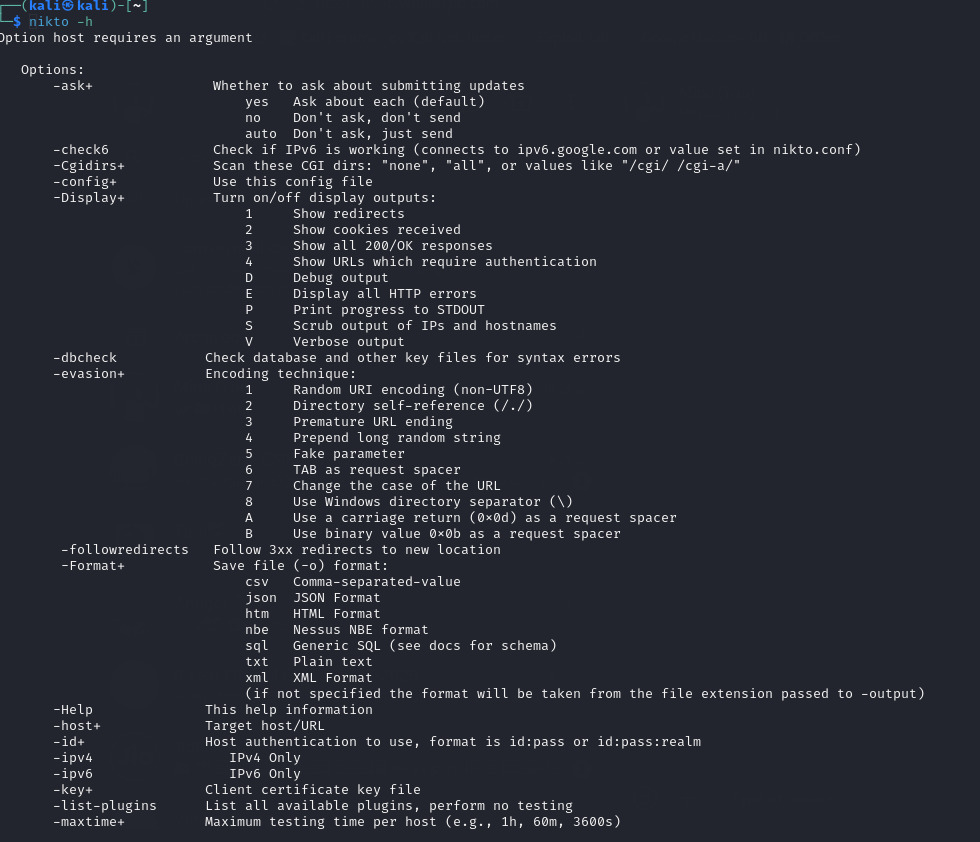
#### IMPLEMENTATION AND DISCUSSION

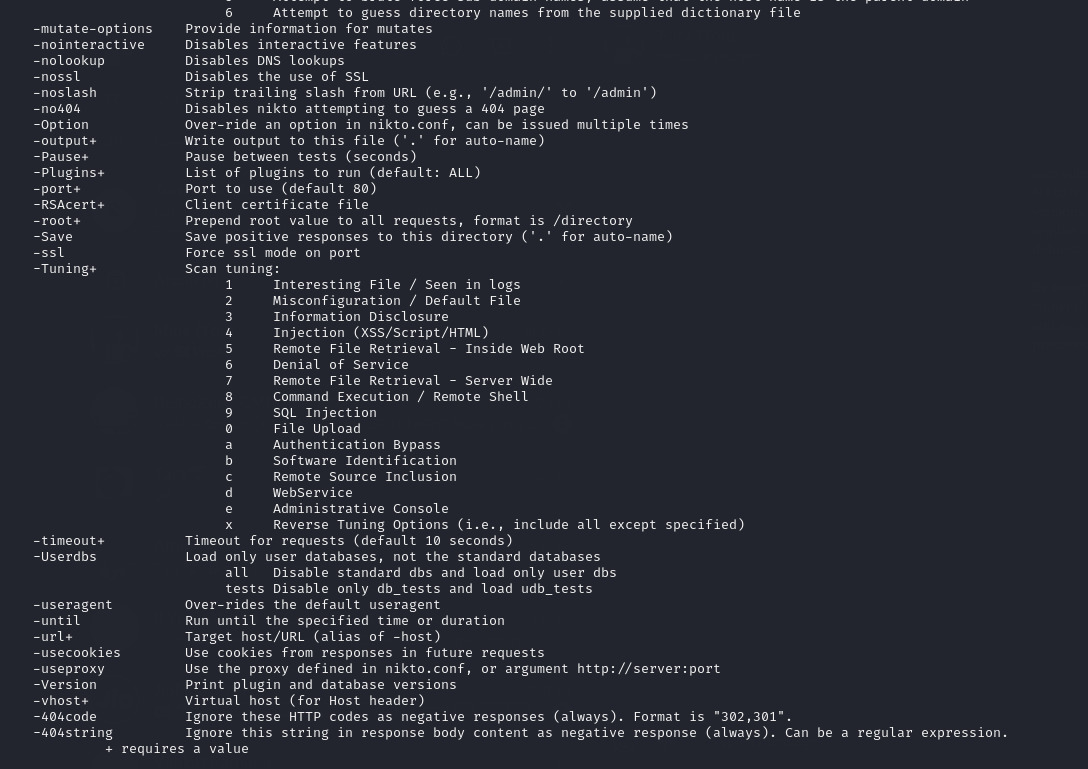
* 1. **IMPLEMENTATION**

**Reconnaissance:**

It is the information-gathering stage of ethical hacking, where you collect data about the target system. This data can include anything from network infrastructure to employee contact details. The goal of reconnaissance is to identify as many potential attack vectors as possible.

### COMMAND: nikto -help

 **Fig 8:** Reconnaissance



## Fig 9: nikto tuning options

## Scanning:

It is the methodical process of inspecting systems, applications, and networks to find any potential flaws, incorrect setups, or vulnerabilities.

### COMMAND: sudo nikto -h <IP address>

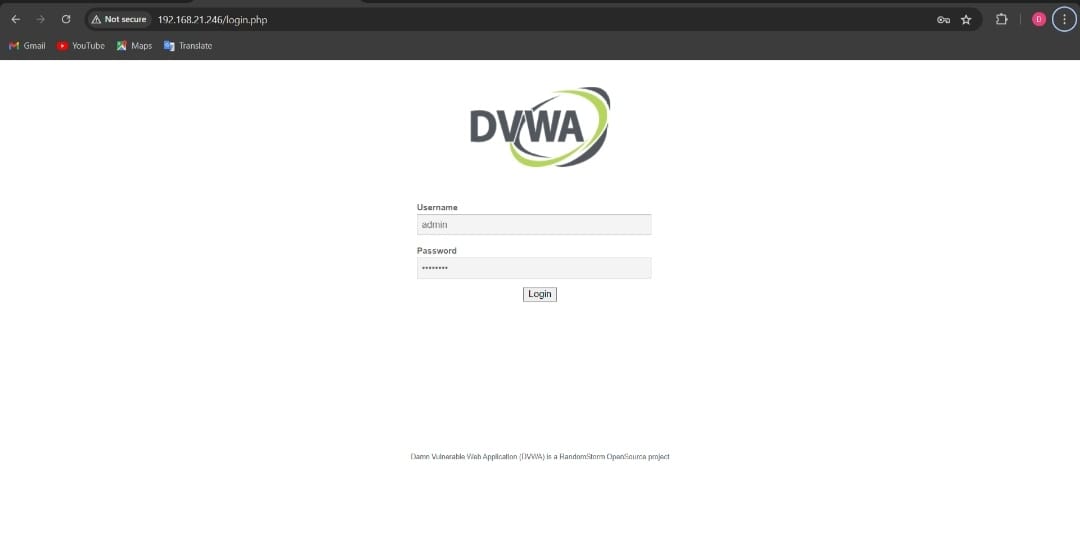
### 

**Fig:10** scanning site

**Command: nikto -h <IP address> -9**



**Fig:11** scanning SQL Injection attack

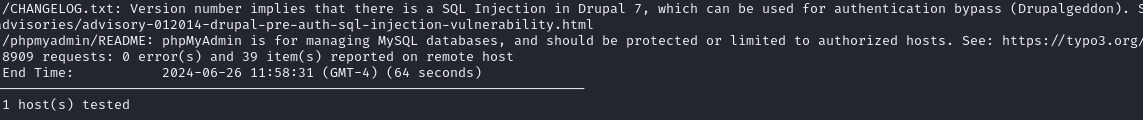


**Fig 12:** DVWA login

# CHAPTER 7

#### RESULT

We have successfully scanned the vulnerability present in the web application which can identify potential vulnerabilities in your site, allowing you to address them before they can be exploited.



**Fig 13**

# CHAPTER 8

#### CONCLUSION AND FUTURE ENHANCEMEMT

##### 8.1. CONCLUSION

Web vulnerability scanning is an effective method to identify potential security issues in web applications. Nikto is a robust, open-source scanner that can detect various vulnerabilities, including outdated software versions, misconfigurations, and potentially dangerous files and scripts. Its comprehensive database and regular updates make it a valuable tool for security professionals looking to bolster their web application defenses. By incorporating Nikto into your security testing regimen, you can proactively identify and address vulnerabilities before they are exploited by malicious actors. This not only helps in maintaining the integrity and security of your web applications but also ensures compliance with various security standards and best practices.

##### FUTURE ENHANCEMENTS

To further enhance the effectiveness of using Nikto for web vulnerability scanning, several future improvements and integrations can be considered:

* **Customizable Scan Profiles**: Allow users to create and use customizable scan profiles tailored to specific web applications or environments. This flexibility can improve the relevance and accuracy of scan results.
* **User-Friendly Interface**: Develop a more intuitive and user-friendly graphical interface for Nikto to make it accessible to users with varying levels of technical expertise. This can encourage wider adoption and more effective use of the tool.
* **Integration with Other Security Tools**: Facilitate seamless integration with other security tools and platforms, such as intrusion detection systems (IDS), security information and event management (SIEM) systems, and other vulnerability scanners. This provides a holistic view of the security posture.

#### REFERENCES

1. Mohammadi, S., & Mirzakuchaki, S. (2017). A Comparative Study of Open Source Web Vulnerability Scanners. Journal of Information Security and Applications, 35, 42-53.
2. Kumar, S., & Sharma, R. (2015). A Study on Web Application Vulnerabilities (SQLIA, XSS) Exploitation and Security Engine for SQL Injection. International Journal of Computer Applications, 120(23).
3. Anusha, A., & Jain, M. (2019). Analysis of Web Application Security Vulnerabilities and Attack Detection Mechanisms. In Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI).
4. Owusu, E., & Metzger, P. (2020). Evaluation of Open Source Web Vulnerability Scanners in Identifying Web Vulnerabilities. In Journal of Cyber Security and Information Systems.