Web Application Vulnerability Scanner

Introduction:

With the increasing number of web-based applications, securing them against cyber threats has become crucial. This project aims to build an automated vulnerability scanner to detect critical security flaws in web applications such as XSS, SQL Injection, and CSRF.

Abstract:

This scanner crawls a web application, detects input fields and forms, and attempts to exploit them using known payloads. It analyzes server responses to identify vulnerable areas. The results are logged, classified by severity, and displayed via a Flask-based web interface. A downloadable HTML report is also generated.

Tools Used:

- Python - for scripting and automation
- Requests - for sending HTTP requests
- BeautifulSoup - for parsing HTML content
- Flask - for building the web interface
- Regex - for pattern matching in responses
- HTML/CSS - for UI and report styling

Steps Involved in Building the Project:

1. **Crawler Module**: Built using BeautifulSoup and Requests, it extracts all forms and links within a given domain.
2. **Payload Injection**: For every form or URL parameter, we inject payloads from files containing XSS, SQLi, and CSRF test cases.
3. **Response Analysis**: Using regex and pattern matching, we detect the reflection of malicious

input and missing tokens.

4. **Vulnerability Classification**: Each detected issue is assigned a severity level based on its impact.

5. **Reporting**: HTML reports are generated with vulnerability summaries, payloads, and affected URLs.

6. **Web UI**: Users can input URLs, start scans, view results in real time, and download the final report.

Conclusion:

This tool helps developers and security researchers detect common vulnerabilities in web applications with minimal manual effort. Its modularity allows easy extension for additional tests and enhancements.