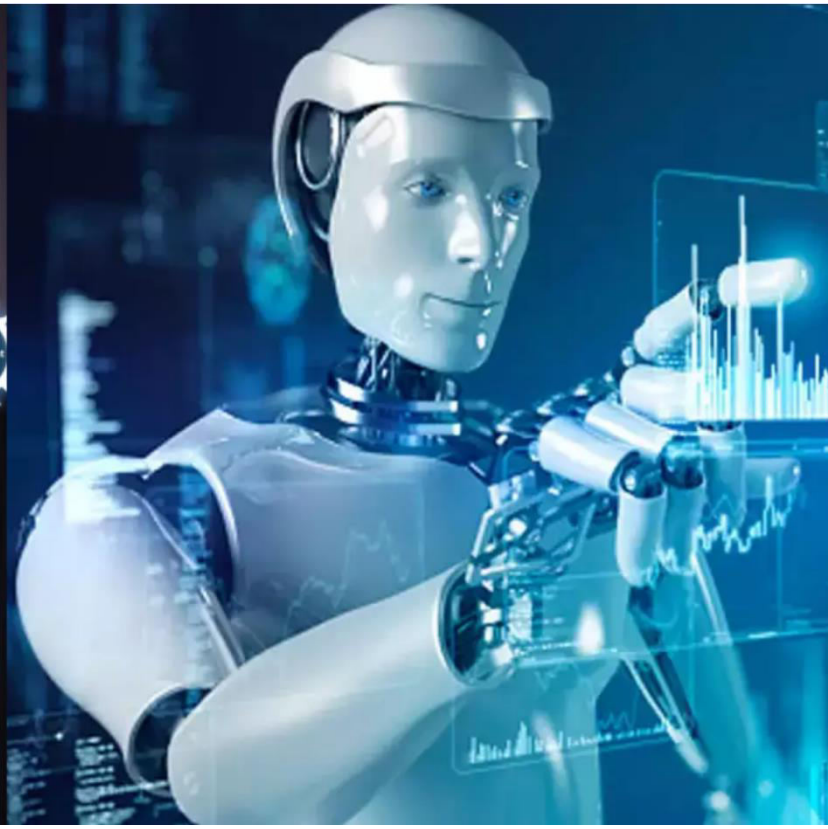
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 15, Issue 3, March 2026

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503164|

CrossGuard: Automated Detection of Cross-Site Scripting (XSS) Vulnerabilities in Web Applications

Prof.S.R. Joshi[1], Santosh Hange[2], Pruthviraj Khaladkar[3], Kartik Kesgire[4], Omkar Patil[5]

Department of Information Technology Engineering, Pimpri Chinchwad Education Trust's, Pimpri Chinchwad Polytechnic, Pradhikaran, Nigdi, Pune, India [1,2,3,4,5]

ABSTRACT: Web applications play an important role in modern digital services such as e-commerce, online banking, social networking, and cloud platforms. However, many web applications contain security vulnerabilities that attackers can exploit. One of the most common web vulnerabilities is Cross-Site Scripting (XSS), which allows attackers to inject malicious scripts into web pages viewed by other users. These scripts execute in the user's browser and may steal sensitive information such as cookies, session tokens, and login credentials.

This research presents **CrossGuard**, an automated XSS vulnerability scanning tool designed to detect Cross-Site Scripting vulnerabilities in web applications. The system is implemented using the Python Flask framework and integrates web crawling, form extraction, payload injection, and response analysis techniques. The scanner automatically explores web pages within a target domain, identifies input forms and URL parameters, injects XSS payloads, and analyzes server responses to detect vulnerabilities. If the injected payload is reflected in the server response, the system reports a potential XSS vulnerability.

The system also generates a detailed scan summary and a downloadable vulnerability report containing information such as vulnerability type, affected URL, payload used, and severity level. The experimental results demonstrate that the CrossGuard system effectively detects XSS vulnerabilities and assists developers in improving the security of web applications.

KEYWORDS: Cross-Site Scripting, Web Security, Vulnerability Scanner, Flask Framework, Web Crawling, Penetration Testing.

I. INTRODUCTION

Web applications have become essential for delivering online services across various industries including education, finance, healthcare, and e-commerce. As the number of web applications increases, security threats targeting these systems have also grown significantly. Attackers frequently exploit vulnerabilities in web applications to gain unauthorized access, steal sensitive data, or disrupt services.

One of the most common vulnerabilities in web applications is **Cross-Site Scripting (XSS)**. XSS occurs when a web application fails to properly validate or sanitize user input. Attackers can inject malicious JavaScript code into webpages, which is then executed in the browser of other users who visit the affected page. These attacks can lead to session hijacking, data theft, website defacement, and unauthorized actions performed on behalf of the victim.

Manual testing for such vulnerabilities can be difficult and time-consuming, especially for large web applications containing multiple pages and input forms. Automated vulnerability scanners help security researchers and developers identify vulnerabilities quickly and efficiently.

This research proposes **CrossGuard**, an automated XSS vulnerability detection tool. The system performs automated web crawling, identifies input points within the application, injects test payloads, and analyzes server responses to detect potential vulnerabilities. By automating the detection process, the tool helps improve the security testing process for web applications.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503164|

II. LITERATURE REVIEW

Several studies have focused on identifying and preventing web application vulnerabilities. Researchers have developed different techniques to detect vulnerabilities such as static code analysis, dynamic analysis, and automated vulnerability scanning.

Security organizations such as OWASP have identified Cross-Site Scripting as one of the most critical vulnerabilities affecting modern web applications. Many vulnerability scanners analyze web applications by injecting malicious payloads into input fields and observing the behavior of the application.

However, many existing vulnerability scanning tools are complex and require advanced configuration. Some tools are also expensive and not easily accessible for small development teams or academic research purposes.

Therefore, there is a need for lightweight and easy-to-use vulnerability scanning tools that can automatically detect XSS vulnerabilities with minimal configuration. The proposed CrossGuard system aims to address this problem by providing a simple and automated solution for detecting XSS vulnerabilities in web applications.

III. METHODOLOGY

The CrossGuard system follows an automated scanning process to detect Cross-Site Scripting vulnerabilities in web applications.

The scanning process begins when the user enters the target website URL in the scanner interface. The system then initiates a web crawling process to discover internal pages within the target domain. The crawler collects links and builds a list of webpages that will be analyzed during the scanning process.

After discovering the webpages, the system analyzes each page and extracts HTML forms and input fields. These input fields represent possible points where attackers may inject malicious input.

The scanner then injects predefined XSS payloads into the input fields and URL parameters. The request containing the payload is sent to the server. After the server responds, the system analyzes the response to determine whether the injected payload appears in the returned webpage.

If the payload is reflected in the server response, the system identifies a potential XSS vulnerability and records the details of the vulnerability.

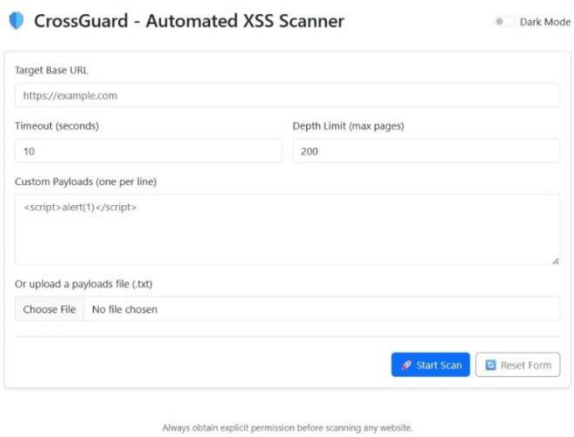


Fig.1 CrossGuard Automated XSS Scanner User Interface

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503164|

IV. EXPERIMENTAL RESULTS

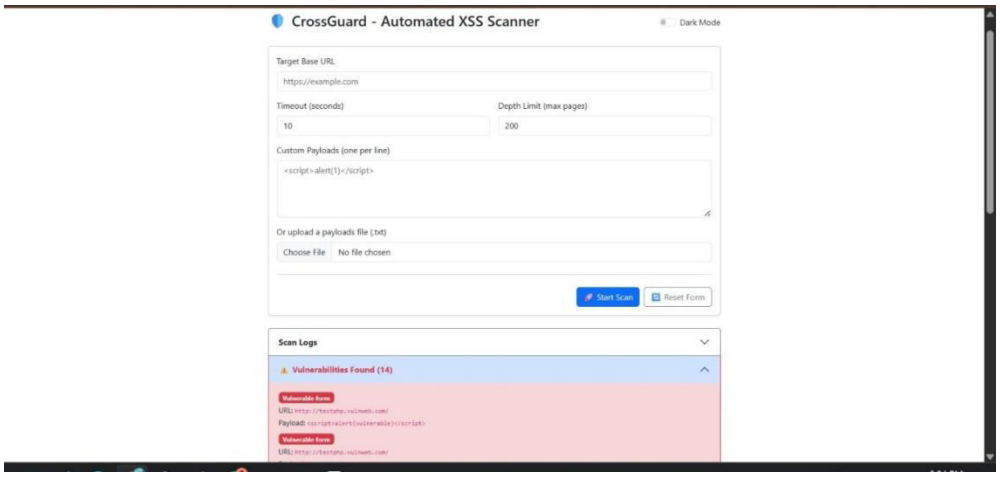


Fig.2 CrossGuard XSS Scanner Vulnerability Detection Result

The figure shows the vulnerability detection result generated by the CrossGuard automated XSS scanner after performing a scan on the target web application. The scanner analyzes the input forms and URL parameters of the web pages by injecting predefined XSS payloads. After submitting the payloads, the system examines the server response to determine whether the injected script is reflected in the webpage.

If the payload appears in the response, the scanner identifies it as a potential Cross-Site Scripting vulnerability. The output section displays important details such as the vulnerable form, affected URL, and the payload used during testing. This information helps security researchers and developers understand where the vulnerability exists and how it can be mitigated.

The detection result also shows the total number of vulnerabilities found during the scanning process, allowing users to evaluate the security status of the target web application.

V. CONCLUSION

Future improvements may include integration with real GPS-based navigation services, advanced analytics for This research presented **CrossGuard**, an automated vulnerability scanning tool designed to detect Cross-Site Scripting (XSS) vulnerabilities in web applications. The proposed system uses automated web crawling, form extraction, payload injection, and response analysis techniques to identify potential XSS vulnerabilities within web pages.

The system analyzes input forms and URL parameters by injecting predefined XSS payloads and monitoring the server response. If the injected payload is reflected in the webpage, the scanner reports it as a possible vulnerability. The generated scan results provide useful information such as affected URLs, payloads used, and vulnerability severity, which helps developers understand the security weaknesses of their web applications.

The experimental results demonstrate that the CrossGuard scanner can effectively detect XSS vulnerabilities and assist developers in improving web application security. By automating the vulnerability detection process, the tool reduces the effort required for manual security testing and increases the efficiency of vulnerability analysis.

In future work, the system can be enhanced by supporting additional web vulnerabilities such as SQL Injection, Cross-Site Request Forgery (CSRF), and other OWASP Top 10 vulnerabilities. Integration with advanced detection techniques and improved reporting mechanisms can further strengthen the effectiveness of the system.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 3, March 2026

|DOI: 10.15680/IJIRSET.2026.1503164|

REFERENCES

[1] OWASP Foundation, "OWASP Top 10 Web Application Security Risks," Available: https://owasp.org
[2] M. Howard and D. LeBlanc, Writing Secure Code, Microsoft Press.
[3] D. Stuttard and M. Pinto, The Web Application Hacker's Handbook, Wiley Publishing.
[4] Flask Documentation, "Flask Web Framework," Available: https://flask.palletsprojects.com
[5] Google Developers, "Python Programming Language Documentation," Available: https://www.python.org
[6] OWASP Foundation, "Cross-Site Scripting (XSS)," Available: https://owasp.org/www-community/attacks/xss

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details