iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** VI **Month of publication:** June 2025

DOI: https://doi.org/10.22214/ijraset.2025.72053

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VI June 2025- Available at www.ijraset.com

An Ideation for a New Web Application Vulnerability Scanner and AI/LLM Enhanced Report Maker

Yash R. Deshmukh[1], Aarti A. Korhale[2], Yashwardhan V. Bhosale[3], Sharmila K. Wagh[4]
Department of Computer Engineering, Modern Education Society's Wadia College of Engineering 19, Late Prin. V.K. Joag Path, Wadia College Campus, Off, Bund Garden Rd, Pune, Maharashtra 411001

Abstract: ***The surge in the use of web applications has increased the number of vulnerabilities that can be exploited by an attacker, making the security of sensitive information paramount to maintain confidence in the users. Conventional vulnerability scanners, though quite useful, tend to be very generalistic and cannot produce specific or customized concise reports that can quicken the response time. A new web application scanner and report-making application are proposed in this article to address this particular problem. This model not only scans for vulnerabilities but also allows for complex reporting that is appropriate for different audiences within the technical and managerial spheres. These include but are not limited to, scanning large areas for places with highly severe flaws, user-friendly, customizable parameters. This tool also has been designed considering the growing demand which requires producing measure report templates in a format compatible with the country's industries' requirements, thus minimizing the compliance and corrective costs. The new system will result in improvement in the overall web application security stress tests increase, reduction in spurious positives and amounts of relevance, and true relevance providing security teams, developers, and project managers comprehend a pragmatic set of tools.***

Keywords: Web Application Security, Vulnerability Scanner, Automated Security Testing, SQL Injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), Directory Traversal, OWASP ZAP, Nmap, Common Vulnerability Scoring System (CVSS), Risk Assessment, Security Reporting, Cyber Threats, Vulnerability Detection, Remediation, Software Development Life Cycle (SDLC), Security Assess- ment Tool, Application Vulnerabilities, Security Posture, Penetration Testing, Vulner- ability Categorization, Real-Time Scanning, Secure Web Applications, Cybersecurity Automation, Security Risk Prioritization.

I. INTRODUCTION

Cyber Security weakness in any organization or even in a small web application can leave devastating results. Established and newly founded organizations focused on improving their light and rapid security solutions. In relation to web applications, it is a highly growing area in terms of assists in development. Unfortunately, this uptake represents the opportunity for attackers, which makes web applications their targets for vulnerabilities and the underlying data. For this Paper, the web application scanner and report maker addresses such issues. Through powerful scanning algorithms, interpreting useful report-generating tools, this kind of tool not only identifies weaknesses but also considers the application reporting of weaknesses easy. Identifying potential vulnerability can significantly cut down security costs and develop development appropriate. Templates of a report maker against industry standards can cut lots of wastes in the organization and enable stress-free regularities for businesses.

II. LITERATURE REVIEW

Sönmez and Kiliç [1] proposed a holistic visualization approach for web application security by integrating multi-project and multi-phase dynamic application security testing results. Their work highlights the importance of visualization techniques in enhancing security assessments across multiple software development projects. Similarly, Lu et al. [2] introduced V-Digger, a scalable vulnerability assessment framework for large-scale ISP networks. Their approach leverages machine learning to improve detection accuracy and enhance security automation in enterprise environments. Vulnerability scanners play a crucial role in identifying security loopholes in web applications. Alazmi and De Leon [3] conducted a systematic literature review on the effectiveness of web vulnerability scanners, concluding that while these tools provide significant benefits, they often miss complex security flaws, requiring complementary testing strategies. Addressing a critical security concern, Zhang et al. [4] developed SQLPsdem, a proxy-

based mechanism for detecting and mitigating second-order SQL injections, a sophisticated attack vector that traditional scanners fail to detect. Penetration testing (VAPT) is another essential component of cybersecurity. Shah and Mehtre [5] proposed a proactive self-defense strategy using VAPT tools, demonstrating how automated vulnerability assessments can strengthen cyber defenses. Softić and Vejzović [6] further explored the impact of VAPT on operating system security, emphasizing that regular penetration testing significantly reduces security risks in enterprise environments. Another study by Shah and Mehtre [7] introduced an automated approach to VAPT using Net-Nirikshak 1.0, a tool designed to streamline penetration testing with improved efficiency. Portable solutions for vulnerability assessment have also gained attention. Pandey et al. [8] proposed a lightweight VAPT implementation that enhances security in resource-constrained environments. Muzammil et al. [9] analyzed web attack vulnerabilities, focusing on Man-in-the-Middle (MITM) attacks and session hijacking, demonstrating how these threats compromise authentication mechanisms in real-time applications.JavaScript-based applications present unique security challenges. Brito et al. [10] conducted a comparative study of JavaScript static analysis tools for detecting vulnerabilities in Node.js packages, concluding that existing tools often struggle with dynamically generated code. Similarly, Qu et al. [11] introduced AdvSQLi, an adversarial SQL injection framework that generates sophisticated attack queries to bypass Web Application Firewalls (WAF). Their findings highlight the limitations of current WAF-as-a-Service solutions in detecting advanced SQL injection attacks. Client-side security is another major concern. Alenzi and Abbase [12] proposed a defensive framework to mitigate Reflected Cross-Site Scripting (XSS) attacks, which remain prevalent in modern web applications. Meanwhile, Amouei et al. [13] developed RAT, a reinforcement-learning-based testing framework that autonomously discovers vulnerabilities in Web Application Firewalls, improving the detection of complex threats.Finally, the evolving landscape of adaptive cyber defense was explored by Shandilya [14], who emphasized the need for intelligent security mechanisms that can dynamically respond to emerging threats. The study highlighted a paradigm shift toward AI-driven security solutions that adapt in real time to evolving cyber risks. these studies collectively demonstrate the continuous evolution of web security technologies, ranging from vulnerability scanners and penetration testing to AI-based security mechanisms. Despite advancements, challenges remain, particularly in detecting sophisticated attacks such as second-order SQL injections, adversarial exploits, and real-time adaptive threats. Future research should focus on integrating machine learning and AI-driven solutions to enhance the accuracy and efficiency of security measures.

III. SYSTEM ARCHITECTURE

The given system architecture diagram represents a scanner-based application that allows users to interact with the system through multiple functionalities. The system is designed to provide a seamless and secure workflow for users, ensuring efficient scanning and report generation.

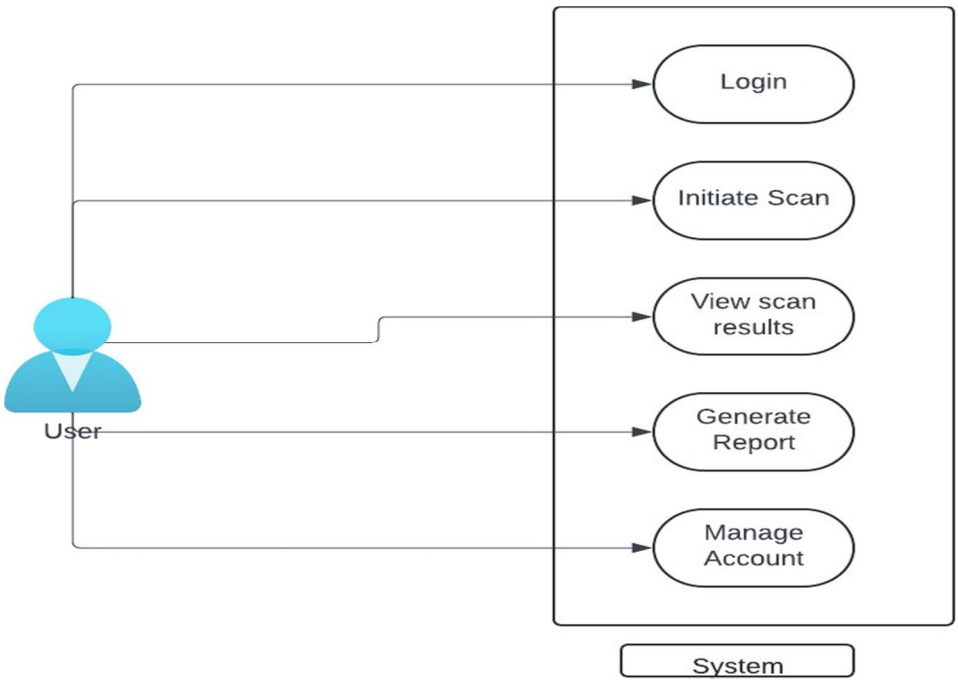


Fig.1. System Architecture Diagram

A. Components of the System

1) User

- The user represents an individual interacting with the system. This can be an administrator, an operator, or an end-user.

2) System (Application Backend & Frontend)

- The system comprises various modules that process user requests and deliver the required outputs.

B. Functional Modules

1) Login:

- The user needs to authenticate themselves before accessing the system. This ensures security and personalized access to functionalities.

2) Initiate Scan:

- Once logged in, users can start a new scan using the scanner system. This module interacts with hardware/software components to capture the required data.

3) View Scan Results:

- After processing, the scanned data is made available for the user to view. The system applies necessary processing techniques, such as image enhancement or recognition, before presenting the results.

4) Generate Report:

- Users can generate detailed reports based on the scanned data. This functionality ensures that the processed information is structured and stored for future use.

5) Manage Account:

- Users can modify their personal information, change settings, and manage access permissions within the system.

IV. RESULTS

The vulnerability scan conducted using the Vulnerability Scanner analysed the security weaknesses of a web application. The scan targeted various critical vulnerabilities, including SQL Injection, Cross-Site Scripting (XSS), Sensitive Data Exposure, and Security Misconfigurations. The results revealed that SQL Injection was the most prevalent vulnerability, accounting for 39.1% of the detected security flaws. This was followed by Data Exposure vulnerabilities at 29.3% and Cross-Site Scripting (XSS) at 28.3%. Other issues, such as CSRF, authentication issues, and security misconfigurations, were also detected but in smaller proportions.

The severity levels of the vulnerabilities were classified into four categories: Low, Medium, High, and Critical. The analysis showed that a significant number of vulnerabilities fell into the High and Critical severity levels, indicating the potential risk of exploitation and the necessity for immediate remediation. These results emphasize the importance of implementing security best practices, including input validation, secure authentication mechanisms, and proper access controls, to mitigate potential threats. The findings provide valuable insights into the security posture of the web application and serve as a foundation for enhancing its overall security framework.

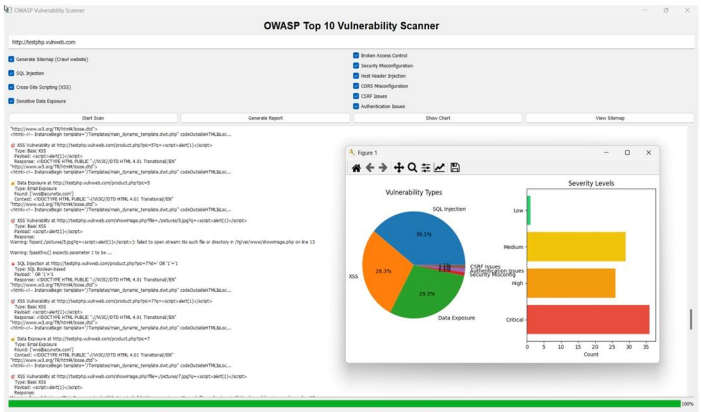


Fig.2. Vulnerability Detected 1

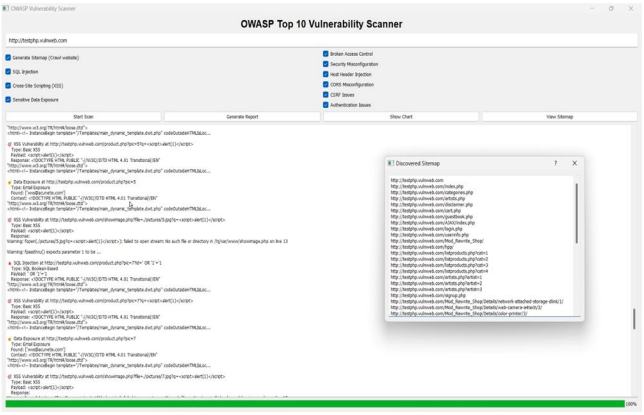


Fig.3. Vulnerability Detected 2

V. CONCLUSIONS

Today, with a surge in web application security threats, it is crucial to incorporate vulnerability management techniques that are both efficient and effective. All of these are advanced assistance systems for organizations that want to improve their security posture, extend their capabilities to the internet, combine detection algorithms with advanced capabilities, and provide customizable user-friendly reporting systems. It enables security teams, developers, and management to have consistent actionable insight which is available in a prioritized, ready for compliance, and understandable manner, improving how fast threats can be found, reported, and fixed.

However, factors such as frequent updates, potential false alerts, and initial configuration can be considered limitations but can be dealt with if appropriate planning and assistance are in place. The scalability of the tool, its compatibility with compliance requirements as well as its capacity to integrate with various platforms conceptualizes the versatility of this tool to suit different organizations regardless of their size or security needs. To conclude, this new web application scanner and report maker is a feasible, reliable, and cost-effective solution meant to enhance web application security in the current era environment. It speeds how fast response actions can be carried out on a threat, lightens the load for security teams thus saving time and effort which, in turn, boosts compliance actions which make web applications safer and stronger than before.

VI. FUTURE WORK

While the current implementation of the Web Application Vulnerability Scanner and Report Maker offers valuable functionality, there are several areas for future work that can further enhance the tool's capabilities, improve its performance, and extend its applicability.

1) Machine Learning and AI Integration:
- Implementing machine learning algorithms can improve vulnerability detec- tion accuracy, particularly in reducing false positives and identifying com- plex attack patterns that traditional methods may miss.
- AI-driven methods could be used for analyzing large amounts of data and providing predictive insights, such as identifying trends in vulnerabilities or recommending remediation actions based on historical data.

2) Performance Optimization:
- Reducing the performance overhead during scans, particularly for large-scale applications, remains an area for improvement.
- Techniques like parallel processing, distributed scanning, or incremental scanning (where only new or changed parts of an application are scanned) could enhance the tool's efficiency.

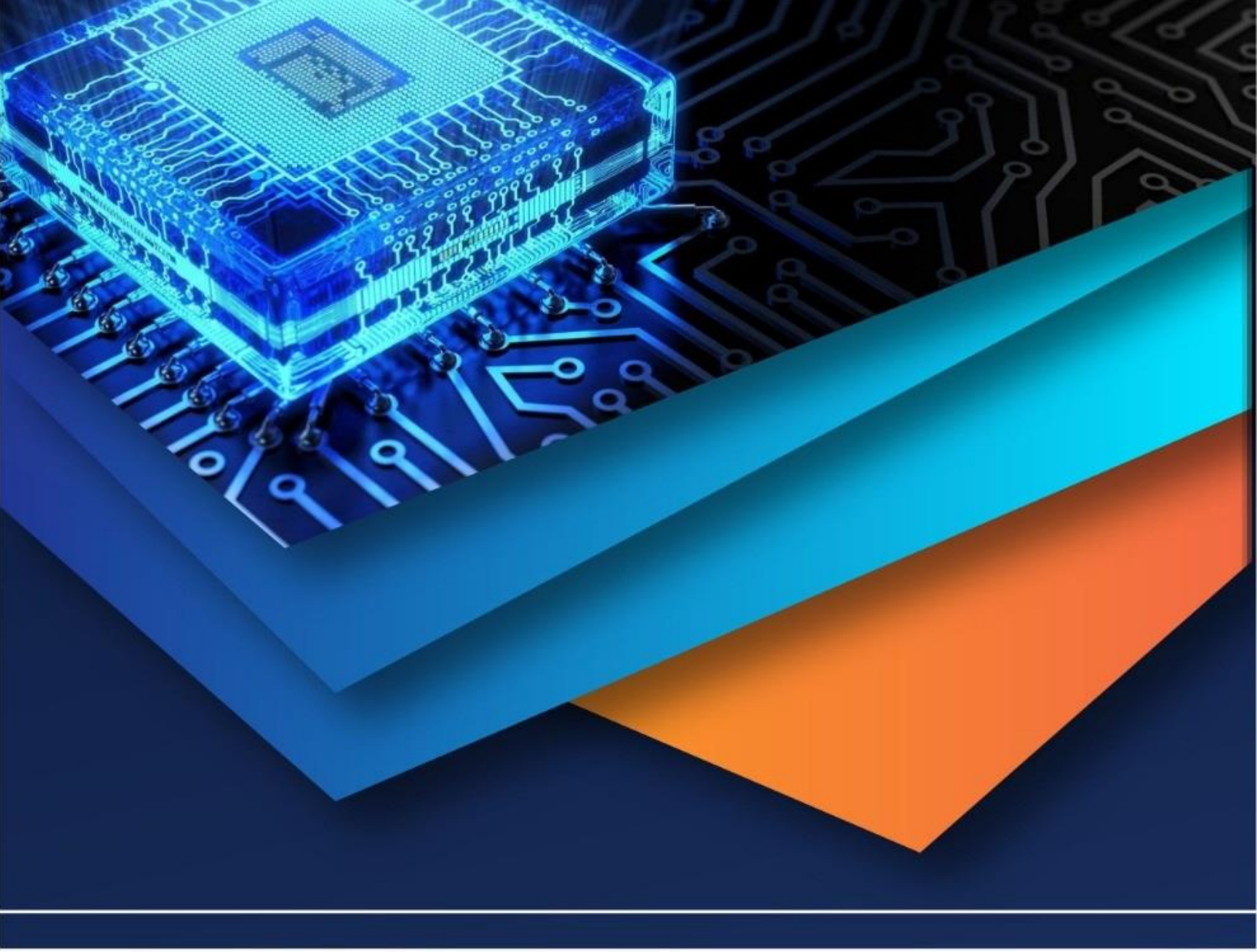
3) User Interface Improvements:
- The user interface could be further optimized to provide a more intuitive experience, with features like drag-and-drop file scanning, real-time scan status updates, and better visualization of vulnerability severity and trends.
- Adding more detailed analytics, such as graphs or heatmaps showing which parts of the application are most vulnerable, could further assist users in prioritizing fixes.

4) Integration with other Security Tools:
- Future versions of the tool could integrate with other security tools, such as firewalls, intrusion detection systems (IDS), or vulnerability management platforms, to provide a more holistic security solution.
- Integrating with a larger security ecosystem could help create a continuous feedback loop for identifying vulnerabilities and addressing them more efficiently.

5) Cloud-Based Deployment
- The tool could be deployed as a cloud service, offering scalability and reducing the need for users to manage their own infrastructure.
- A cloud version would allow for real-time updates, easier integration with other cloud-based tools, and enhanced collaboration among teams in large organizations.

REFERENCES

[1] F. Ö. Sönmez and B. G. Kiliç, "Holistic Web Application Security Visualization for Multi-Project and Multi-Phase Dynamic Application Security Test Results," IEEE Access, vol. 9, pp. 25858–25884, 2021, doi: 10.1109/ACCESS.2021.3057044.
[2] N. Lu, R. Huang, M. Yao, W. Shi, and K.-K. R. Choo, "V-Digger: An Efficient and Secure Vulnerability Assessment for Large-Scale ISP Network," IEEE Transactions on Dependable and Secure Computing, vol. 21, no. 4, pp. 3227–3246, July-Aug. 2024, doi: 10.1109/TDSC.2023.3324646.
[3] S. Alazmi and D. C. De Leon, "A Systematic Literature Review on the Characteristics and Effectiveness of Web Application Vulnerability Scanners," IEEE Access, vol. 10, pp. 33200–33219, 2022, doi: 10.1109/ACCESS.2022.3161522.
[4] B. Zhang, R. Ren, J. Liu, M. Jiang, J. Ren, and J. Li, "SQLPsdem: A Proxy-Based Mechanism Towards Detecting, Locating and Preventing Second-Order SQL Injections," IEEE Transactions on Software Engineering, vol. 50, no. 7, pp. 1807–1826, July 2024, doi: 10.1109/TSE.2024.3400404.
[5] S. Shah and B. M. Mehtre, "A reliable strategy for proactive self-defence in cyberspace using VAPT tools and techniques," 2013 IEEE International Conference on Computational Intelligence and Computing Research, Enathi, India, 2013, pp. 1–6, doi: 10.1109/ICCIC.2013.6724216.
[6] J. Softić and Z. Vejzović, "Impact of Vulnerability Assessment and Penetration Testing (VAPT) on Operating System Security," 2023 22nd International Symposium INFOTEH-JAHORINA (INFOTEH), East Sarajevo, Bosnia and Herzegovina, 2023, pp. 1–6, doi: 10.1109/INFOTEH57020.2023.10094095.
[7] S. Shah and B. M. Mehtre, "An automated approach to Vulnerability Assessment and Penetration Testing using Net-Nirikshak 1.0," 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies, Ramanathapuram, India, 2014, pp. 707–712, doi: 10.1109/ICACCCT.2014.7019182.
[8] R. Pandey, V. Jyothindar, and U. K. Chopra, "Vulnerability Assessment and Penetration Testing: A portable solution Implementation," 2020 12th International Conference on Computational Intelligence and Communication Networks (CICN), Bhimtal, India, 2020, pp. 398–402, doi: 10.1109/CICN49253.2020.9242640.
[9] M. B. Muzammil, M. Bilal, S. Ajmal, S. C. Shongwe, and Y. Y. Ghadi, "Unveiling Vulnerabilities of Web Attacks Considering Man in the Middle Attack and Session Hijacking," IEEE Access, vol. 12, pp. 6365–6375, 2024, doi: 10.1109/ACCESS.2024.3350444.
[10] T. Brito et al., "Study of JavaScript Static Analysis Tools for Vulnerability Detection in Node.js Packages," IEEE Transactions on Reliability, vol. 72, no. 4, pp. 1324–1339, Dec. 2023, doi: 10.1109/TR.2023.3286301.
[11] Z. Qu, X. Ling, T. Wang, X. Chen, S. Ji, and C. Wu, "AdvSQLi: Generating Adversarial SQL Injections Against Real-World WAF-as-a-Service," IEEE Transactions on Information Forensics and Security, vol. 19, pp. 2623–2638, 2024, doi: 10.1109/TIFS.2024.3350911.
[12] K. F. Alenzi and O. A. Bashir Abbase, "A Defensive Framework for Reflected XSS in Client-Side Applications," Journal of Web Engineering, vol. 21, no. 7, pp. 2209–2229, October 2022, doi: 10.13052/jwe1540-9589.2179.
[13] M. Amouei, M. Rezvani, and M. Fateh, "RAT: Reinforcement-Learning-Driven and Adaptive Testing for Vulnerability Discovery in Web Application Firewalls," IEEE Transactions on Dependable and Secure Computing, vol. 19, no. 5, pp. 3371–3386, 1 Sept.-Oct. 2022, doi: 10.1109/TDSC.2021.3095417.
[14] S. K. Shandilya, "Paradigm Shift in Adaptive Cyber Defense for Securing the Web Data: The Future Ahead," Journal of Web Engineering, vol. 21, no. 4, pp. 1371–1376, June 2022, doi: 10.13052/jwe1540-9589.2141.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY