AI-BUG-BOUNTY: AN AUTOMATED VULNERABILITY SCANNER USING ARTIFICIAL INTELLIGENCE

Project Reference No.: *49S_BE_6032*

College : R L Jalappa Institute Of Technology, Doddaballapura, Bengaluru Rural
Branch : Department of CS&E (Artificial Intelligence and Machine Learning)
Guide(s) : Mrs. Anupriya A G
Dr. Manjunatha B N
Student(s) : Ms. Vishnupriya N
Mr. Jagathi Charan Kumar
Ms. keerthika S
Ms. Rashmi E

Keywords:
Artificial Intelligence, Web Vulnerability Detection, Bug Bounty Automation, Web Application Security, Machine Learning, NLP-based Explanation, XSS Detection, SQL Injection Detection, Automated Security Scanning, Severity Classification, Security Report Generation, Cybersecurity Automation, OWASP Top 10, AI in Cybersecurity.

Introduction:

1. Today, web applications are used everywhere — in banking, healthcare, education, shopping, and even government services. As these applications grow more complex, keeping them secure becomes harder. If security weaknesses are left unnoticed, they can lead to serious problems like data theft, financial loss, identity misuse, and service failure. This makes finding and fixing vulnerabilities very important.
2. Bug bounty programs allow ethical hackers to help companies find security issues before attackers do. However, traditional methods like manual testing and code review take a lot of time, cost more, and depend heavily on the tester's skills. Automated tools help speed up the process, but they often give false alerts and fail to detect new or logical vulnerabilities.
3. To solve these problems, this project introduces AI-Bug-Bounty: An Automated Vulnerability Scanner Using Artificial Intelligence. This system uses AI and Machine Learning to automatically detect common web application issues such as XSS, SQL Injection, authentication flaws, and configuration errors. It can scan a single website or multiple sites using a CSV file, making it suitable for large-scale analysis.
4. One of the key features of this system is that it provides simple, easy-to-understand explanations of the detected issues and ranks them based on their severity. This reduces manual effort, improves security analysis, and makes vulnerability scanning easier for students, beginners, and bug bounty learners.

Objectives:

1. Build an automated web vulnerability scanner that checks website URLs without manual effort.

2. Use AI (OpenAI GPT models) to intelligently detect vulnerabilities and explain them in simple terms.
3. Identify common security issues such as XSS, SQL Injection, CSRF, Clickjacking, misconfigurations, and data exposure.
4. Generate structured security reports with severity levels, remediation steps, PDF export, and scan history.
5. Provide a fast, user-friendly dashboard that helps students, developers, and bug bounty learners understand and fix website weaknesses easily.

Methodology:

1. User Access (Frontend Layer) User opens the web app in a browser. Enters a single URL or uploads a CSV file of URLs.
2. Request Handling (Frontend → Backend) React + Vite frontend sends the request to the backend via HTTP API.Input is validated before processing.
3. URL Processing (Backend Layer) Node.js + Express server receives the URL(s). Fetches webpage content (HTML, headers, status, forms, links, scripts).
4. Data Preparation for AI Extracted data (HTML snippet, forms, scripts, metadata) is structured. A detailed prompt is built for the OpenAI model.
5. AI-Based Vulnerability Analysis (OpenAI Layer) OpenAI GPT model analyzes the response data. Identifies possible vulnerabilities and assigns severity. Generates explanations and remediation suggestions.

Result and Conclusion:

Result:

The AI-Bug-Bounty system was tested with both vulnerable and safe website URLs, including bulk scans through CSV files. The results showed that the system could accurately detect common web vulnerabilities such as XSS, SQL Injection, input validation issues, and insecure configurations with very few false positives. It successfully classified issues into Low, Medium, High, and Critical levels and generated clear AI-based explanations for each vulnerability. The automated PDF reports were well-structured and useful for documentation and learning. Overall, the system proved to be fast, reliable, easy to use, and more informative compared to traditional vulnerability scanners.

Conclusion:

The AI-Bug-Bounty system demonstrates how Artificial Intelligence can improve web vulnerability detection by automating the scanning process, reducing manual effort, and providing easy-to-understand explanations with severity prioritization. It effectively detects common security issues, supports both single and bulk URL scanning, and generates professional reports. The system is especially helpful for students, beginners, and bug bounty learners. Although it currently focuses on common vulnerabilities, it provides a strong foundation for future improvements like real-time scanning and advanced threat detection.

References:

1. [1] D. Cotroneo, R. De Luca, and P. Liguori, “DeVAIC: A Tool for Security Assessment of AI-Generated Code,” arXiv preprint arXiv:2404.07548, 2024.
2. [2] J. Zheng, et al., “From Reviewers’ Lens: Understanding Bug Bounty Report Invalid
3. Reasons with Large Language Models,” arXiv preprint arXiv:2511.18608, 2025.
4. [3] S. Shimmi, H. Okhravi, and M. Rahimi, “AIBased Software Vulnerability
5. Detection: A Systematic Literature Review,” arXiv preprint arXiv:2506.10280, 2025.
6. [4] OWASP Foundation, “OWASP Top 10 Web Application SecurityRisks,” OWASP, 2023.
7. [Online]. Available: https://owasp.org
8. [5] OWASP Foundation, “OWASP Top 10 for Large Language Model Applications,” OWASP, 2024. [Online]. Available: https://owasp.org
9. [6] OWASP Foundation, “Machine Learning SecurityTop 10,” OWASP Project, 2023.
10. [Online]. Available: https://owasp.org/www-
11. [7] HackerOne, “AI Security Bug Bounty Engagement Best Practices,” HackerOne Help
12. Center, 2024. [Online]. Available: https://docs.hackerone.com
13. [8] Bugcrowd, “OWASP Top 10 Security Threats Facing AI Systems,” Bugcrowd Blog, 2023. [Online]. Available: https://www.bugcrowd.com/blog/
14. [9] NIST, “AI Risk Management Framework,” National Institute of Standards and Technology, 2023. [Online]. Available: https://www.nist.gov
15. [10] MITRE Corporation, “MITRE ATLAS: Adversarial Threat Landscape for AI Systems,” 2023. [Online]. Available: https://atlas.mitre.org
16. [11] ENISA, “Artificial Intelligence Threat Landscape,” European Union Agency for Cybersecurity, 2023. [Online]. Available: https://www.enisa.europa.eu
17. [12] IEEE, “Artificial Intelligence and Cybersecurity,” IEEE Xplore Digital Library, 2024.
18. [Online]. Available: https://ieeexplore.ieee.org
19. [13] OWASP Foundation, “OWASP GenAI Security Project – Top Risks for Agentic AI,” OWASP, 2025. [Online]. Available: https://genai.owasp.org project-machine-learning-security-top-10/
20. [14] Huntr Security, “Bug Bounty Programs for AI and Machine Learning,” Huntr Platform, 2024. [Online]. Available: https://huntr.com
21. [15] GitHub Repository, “AI-Bug-Bounty: Automated Vulnerability Scanner Using AI,” 2024.
22. [Online]. Available: https://github.com

Future Scope:

1. Enable **real-time vulnerability scanning** to continuously monitor websites for new threats.
2. 2.Improve detection of **advanced and zero-day attacks** using more powerful AI and deep learning models.

3. 3.Deploy the system on the **cloud** for better scalability and performance across many websites.
4. Integrate with popular **bug bounty platforms** for automatic vulnerability reporting.
5. Expand support to detect more issues such as **CSRF, SSRF, authentication bypass, and authorization flaws**.
6. Add **multilingual AI explanations** to make reports accessible to a wider audience.
7. Provide **advanced dashboards and visualizations** to track security trends and scan history.
8. Continuously **retrain AI models** with new vulnerability data to improve accuracy and reduce false alerts.