RESEARCH ARTICLE OPEN ACCESS

Intelligent Web Security Systems: A Literature Survey on AI-Driven Vulnerability Detection and Automated Penetration Testing

Tharun G
Department of ISE
Don Bosco Institute of Technology
Bengaluru, India
tharung189@gmail.com

Kiran Kumar
Assistant Professor
Department of ISE
Don Bosco Institute of Technology
Bengaluru, India
kirandp45@gmail.com

Sankeerth S
Department of ISE
Don Bosco Institute of Technology
Bengaluru, India
sankeerth1920@gmail.com

Rohan M
Department of ISE
Don Bosco Institute of Technology
Bengaluru, India
rohandvg2005@gmail.com

Thanushree Patel B S
Department of ISE
Don Bosco Institute of Technology
Bengaluru, India
thanuguna9110@gmail.com

***Abstract*—** Web Applications are an integral part of today's technological age, carrying out various applications in areas like healthcare, financial institutions, education, e-commerce, and government. Meanwhile, web application development has created a growing number and variety of cyber threats, with some of the most common being SQL injection (SQLi), XSS, CSRF, authentication bypasses, and more. The traditional cybersecurity mechanisms (e.g., vulnerability scanner tools, penetration testing) have proven to be inadequate, however, with respect to multiple factors that are inherent in the existing web environment, such as false positives, non-adaptability, lack of scalability, or a high level of human interaction. In this context, the recent developments in Artificial Intelligence (AI), machine learning (ML), Large Language Models (LLMs), and intelligent penetration testing (IPT) brought several novel solutions to the challenge of web application security and enabled intelligent and self-adaptive security systems. This paper, therefore, reviews the literature on the most recent developments in the area of intelligent web security, such as AI vulnerability detection, intelligent penetration testing, intelligent scanners and multi-agent systems, and the application of LLMs in cybersecurity. This paper aims to provide an introduction to the AI-powered web security system and discuss the need for adaptive and scalable and intelligent cybersecurity approaches to enhance the effectiveness of vulnerability detection.

Keywords—Research papers: Artificial Intelligence, Web Application Security, Vulnerability Discovery, Automatic Penetration Testing, Machine Learning, Large Language Models (LLM), Cyber, Security, SQL Injection, Cross Site Scripting (XSS), Ethical Hacking, Security Automation, Multi-agent, Vulnerability Assessment.

I. INTRODUCTION

With a huge variety of critical services—including financial, healthcare, education, e-commerce, and even government infrastructure—now being delivered online, modern web applications are a vital component in the digital ecosystem. As these critical services proliferate, so do the number of attacks that can be detected and prevented with enhanced cyber security mechanisms, such as SQL Injection (SQLi), Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), and authentication bypassing attacks. Traditional techniques such as web security scanners that are used to detect potential vulnerabilities and penetration testers that are used to test the web for vulnerabilities tend to have high false-positive rates and lack human involvement to be effective. The rise of Artificial Intelligence (AI), Machine Learning (ML), and Large Language Models (LLMs) and their applications across various industries have given rise to the creation of web security measures that leverage AI to automate vulnerability assessment, penetration testing, and enhanced security analysis processes. The purpose of the present literature survey is to review the latest studies in the related field, to overview the current approaches and the direction of the research, to review the existing challenges and limitations of the state of the art solution and to recommend for further improvement.

A. Challenges in web application security

Web Applications are continuously targeted by new generation of cyber attackers. Therefore, many organisations and software developers are concerned about the security of web applications. SQL injection, XSS, CSRF, insecure application programming interfaces (API), broken authentication, and others are the various kinds of cyber-attack. The attacks bring different challenges to the businesses, such as unauthorised access to a company's database, data theft and financial loss [1] [3]. Existing conventional vulnerability detection methods including vulnerability scanners and penetration testing tools have a number of drawbacks, such as false positives, need special skills, and are not capable of detecting complex or zero day attacks [2] [4]. Besides, the dynamism of Web applications and the sophistication of cyber-attack introduce new challenges to the developers and analysts to secure the Web applications [5] [6].

B. Application of Web Security

The advancement of Artificial Intelligence (AI) has seen a rise in the number of innovative Web Application Security solutions that can offer autonomous, self-learning, and self-adaptive protection method [2] [4]. Security software, such as modern security systems, utilizes ML, DL, LLMs, and intelligent agents to detect vulnerabilities, analyze attack methods, conduct penetration tests, and provide detailed reports. The techniques are used for effective vulnerability assessment and to minimize the time spent on manual assessment [5] [7].

AI-based scanners can also learn and update their system from previous data, identify intricate patterns, and adjust to new attack techniques, unlike the rule-based algorithms of traditional scanners.

adaptive scanners, multi-agent frameworks, AI-enhanced validation, and risk prioritization techniques have been implemented to improve the accuracy of vulnerability detection and reduce false-positives [8] [10]. Overall, it can be noted that the combination of AI and modern web technologies enables the development of effective and highly-efficient web security solutions [11] [12] [14].

C. Key Technologies

The technologies showcased within this survey are AI, Machine Learning (ML), Deep Learning (DL), Large Language Models (LLMs), and Multi-Agent Systems for intelligent security testing of Web Apps, and have been used to enhance Web App security [2] [4] [7]. Security scanners based on these technologies can detect, analyze and validate potential threats, and can more accurately and efficiently assess risks associated with such threats than traditional rule-based scanners [5] [8]. Furthermore, this survey includes modern web security technologies and frameworks that are used to create intelligent scanners, such as automated vulnerability scanners, adaptive penetration testing frameworks, static, dynamic, and interactive application security testing solutions, REST API security scanners, and AI-based vulnerability exploit validation technologies [1] [6] [9]. The terms auto-security analysis, penetration testing plan development, report generation and assistance for security experts are also discussed in this survey, in relation to LLMs [10] [11].

Finally, This research gives an overview of some of the technologies that can be used to develop scalable intelligent web security systems such as cloud security systems, real-time monitoring systems, adaptive scanners with feedback loops, and risk assessment tools [3] [12]. All these technologies play an important role in the design of an effective and scalable intelligent web security system that can provide security to modern web applications in the presence of a number of network threats while reducing the human intervention for the security audit process [13] [14].

D. Research Questions

This survey aims to answer:

• How can the effectiveness of AI-based web security solutions for detecting and mitigating vulnerabilities, like SQL Injections (SQLi), Cross-site Scripting (XSS) and authentication weaknesses, be judged? [1] [3] [5]

• What is the added value of using Artificial Intelligence (AI), Machine Learning (ML), Deep Learning (DL), and Large Language Models (LLMs) for vulnerability identification and penetration testing, versus traditional security solutions? [2] [4] [7]

- What are the main disadvantages of the existing generation of Web security solutions built with AI technologies in terms of identification accuracy, false positive rates, scalability, adaptability, explainability, etc.? [6] [9] [11]
- What is the best way for organizations to apply AI, Intelligent Automation, and Adaptive Security Architecture to develop effective and scalable solutions to secure modern Web Apps? [8] [10] [12] [14]

E. Contributions

- Extensive overview of the current AI solutions in web application security and penetration testing frameworks, and intelligent vulnerability discovery [1] [3] [5].
- An analysis and comparison of the usage of AI, machine learning, deep learning, LLMs and multi-agent systems for building automated security evaluation and vulnerability detection mechanisms [2] [4] [7].

high false positive rates, lack of adaptability, web application scalability and integration problems with existing security solutions [6] [9] [11].

- In the last few years, AI enhanced penetration testing, adaptive scanners, and exploit validation and security risk assessment [8] [10] [12] were reviewed.
- Using this survey, the design of scalable and intelligent web security system that can increase the accuracy of vulnerability detection and decrease the overhead of security evaluation , and eventually increase the level of security in web applications[13][14] are addressed

F. Paper Organization

The rest of this survey paper is organized as follows. The survey and comparison of the state-of-the-art techniques related to AI-based web security systems, automated penetration testing, and intelligent techniques for vulnerability assessment are presented in Section II. The third section discusses the essential parameters, architecture, process flow and enabling technologies of intelligent web security system. Section IV discusses the challenges in AI-based web security, which include scalability, detection rate, false positive and security automation. Section V focuses on the research gap, future directions, comparative analysis, and results from the review of the literature. Finally, in section VI [1] [4] [8] [12] conclusions and future research directions on Web Application Security and Automated Vulnerability Assessment using Artificial Intelligence are shared.

II. KEY PARAMETERS OF AI-BASED WEB SECURITY SYSTEMS

AI-driven web security solutions provide intelligent algorithms, automated security analysis, and adaptive threat detection to protect today's web apps from any cyber attack [2][4]. These platforms leverage artificial intelligence, machine learning, large language models and automated penetration testing to detect, confirm, and assess threats and vulnerabilities. AI-based security systems could learn from attacks and provide proactive cyber security, in contrast to conventional security systems.

- Artificial Intelligence & Machine Learning: An automated scanning and penetration test (ATP) identifies SQL injection (SQLi), cross-site scripting (XSS), cross-site request forgery (CSRF), authentication problems, and insecure APIs in a web application automatically, through the use of AI security solutions [1] [3].
- Adaptive Security Assessment: The modern scanners, which are based on AI, provide adaptive and continuous application security assessment, automatically update and adjust the assessment to new threats, prioritize detected flaws according to the assigned risks and recommend the appropriate remediation [4] [8].
- Web Security Technologies: Intelligent web security solutions, generally, employ Static Application Security Testing (SAST), Dynamic Application Security Testing (DAST), REST API security testing, code validation and cloud-based monitoring tools for successful vulnerability assessment and exploitation [5] [9].

security tools can produce detailed reports about the security state of an organization, rank risk levels to the discovered vulnerabilities based on common CVSS standards and suggest automated vulnerability remediation [10] [11]. The ability to scale and automate intelligently

- Scalability and Intelligent Automation. AI powered: web security solutions have been developed to provide automated and ongoing security assessment, monitoring, mass scan and decision making regarding cyber security in more precise and proactive manner [5] [7].

Table I: Key Parameters and Their Benefits in AI-Based Web Security Systems

Key Parameter	Implementation Example	Expected Benefit
Vulnerability Detection	AI-based vulnerability scanner	Detects security flaws automatically
Machine Learning	ML & Deep Learning models	Improves detection accuracy
Large Language Models	AI-assisted security analysis	Intelligent vulnerability assessment
Automated Penetration Testing	AI-based testing framework	Reduces manual testing effort
Risk Assessment	CVSS scoring & AI reports	Better vulnerability prioritization
Security Monitoring	Continuous threat monitoring	Real-time attack detection
Scalability	Cloud-based architecture	Supports large-scale applications

- Assessment & Reporting: Contemporary web security solutions, that are based on AI, offer a longer vulnerability assessment report, analysis of risks, and suggestions for problems fixed as a result of identified vulnerabilities [4] [7]. These solutions rely on the use of traditional metrics for prioritizing the issues and vulnerabilities identified, risk analysis, and assisting the security professionals in the fixed immediately. The AI-based reporting has lowered the number of false positives by checking potential issues before giving the security report [11].

- Scalability Challenges: Scalability and detection accuracy are one of the important concerns when it comes to developing AI-powered web protection solutions [5] [9]. Web applications see huge traffic volumes and have to be updated regularly, which means that security software must run 24 hours a day. Meanwhile, there are a large number of applications, and the requirement for real-time vulnerability analysis which brings in questions about the cost and performance of these systems [10] [12] [14].

III. SYSTEM ARCHITECTURE AND KEY TECHNOLOGIES.

A. Overall Architecture

The proposed Intelligent Web Security System is supposed to automatically detect vulnerabilities in a web application, simulate a penetration test, and generate the reports about the security state of the application. The system adopts the method of Artificial Intelligence (AI), Machine Learning (ML), Large Language Models (LLMs) and automated web security assessment for intelligent analysis of the security of the system [2][4]. It brings together various modules and functions, which are connected securely and automatically assess, decide and monitor the security of Web applications [5][8].

B. The Central Processing Unit is known as Backend.

The central processing unit, responsible for all communication links between the other modules [3] [6] is the Backend Security Engine. The back end handles the user authentication, vulnerability scan initiation, processing of AI models, exploit testing, generation of reports, database, and API operations. The back end tasks the processing of information involved in the functions described above and carries out the processing automatically and in a secure manner in the system [7] [10].

C. User Interaction

The system is accessed by users through the web interface designed for security analysts, developers and administrators [1] [5]. Users can use the web interface to perform vulnerability scan and customize the system for security assessment, track the scan progress, view the detected vulnerability details and create the reports on security issue [8] [11]

D. Functional Modules

The architecture consists of several functional modules that are involved in the process of analyzing the website's security [2] [4]. The components of these modules are: User Management, Vulnerability Scanning, and AI Analysis Engine. The first part handles the registration, authentication and authorization of users. The second one performs a scan of the application for vulnerabilities like SQL injection (SQLi), cross-site scripting (XSS), cross-site request forgery (CSRF), authentication problems, and insecure application programming interfaces (APIs). The final part of the architecture is used to support the identified vulnerabilities, prioritize the detected issues [6] and to decrease false positives [9]

E. Database Management

The database will store information about users the results from vulnerability scans, security logs, a collection of knowledge about vulnerabilities the results from intelligence processing and security reports [5] [7]. Good database management is important to make sure that information is stored correctly data about security can be

looked at. Also having everything in one place makes it easier to watch the system all the time and plan, for security checks [10] [12].

F. AI and Machine Learning Integration Artificial

Intelligence and Machine Learning are the parts of the solution that is being suggested [2] [8]. Both machine learning algorithms and Large Language Models look at weaknesses that are found figure out how attacks might happen check if there are ways to use those weaknesses sort out the risks that are discovered and give ideas on how to deal with them [9] [11].

G. REST API Integration

We need to integrate REST APIs so that the user interface and the backend service and the AI components and the vulnerability scanners and the database all work together smoothly. The REST APIs help us exchange information in a way. They also help us do vulnerability assessments and work with security tools and services.

H. Data Flow

It all starts when we submit a web application to be checked for security issues through a web portal. Then the backend gets the request. Sends it to the vulnerability scanning module to find any potential security issues with the web application. After that we check the issues we found to make sure they are real. We figure out which ones are the most important based on how much of a risk they are. We use AI models to test if the issues can actually be used to hurt the web application. Then we store all the information in the database. Finally the reporting module gives the people using the system a report, on the security issues we found in the web application.

I. Scalability

The system is designed to handle checks for weaknesses big scans of web application security and real-time security watching. This is possible because the system is made up of parts so new artificial intelligence models, security scanners and check modules can be added without affecting the other parts. Using cloud services simple APIs and automated intelligence will make this work.

IV. CHALLENGES IN AI-DRIVEN WEB SECURITY AND AUTOMATED PENETRATION TESTING

A. Limitations of Traditional Web Security Approaches

now web application security testing usually involves scanners that look for known weaknesses and people doing penetration testing to find weak points in web applications. These methods are common and work well but they have some big problems. They cannot find complex weaknesses they give a lot of false warnings and they need very skilled people to look at the results. Also traditional web application security solutions are not good, at handling threats and technologies so they become outdated.

B. Dataset Quality and Availability

The AI-based cybersecurity system needs to have a lot of different data sets to find and validate vulnerabilities and threats. This is really important. At the time researchers say that it is hard to make complete data sets for AI-based security software. This is because there is not data people are worried about privacy the databases are old and the cybersecurity environment is changing really fast. So the quality of the data that goes into the AI-based software is very important. If the data set is not correct the software will not be able to find vulnerabilities and will not work well. That is why researchers think that making data sets is a very important area of research for cybersecurity.

C. Scalability and Real-Time Security Analysis

The AI-based software should be able to handle a lot of work and support real-time security analysis. This is so that web applications can work efficiently stay safe and find threats all the time. However when there are users and services and more API calls and potential security threats it gets harder to respond quickly to threats without making the application work slower. At the time researchers are trying to make AI-based security systems better so they can process data in real-time and analyze it quickly. There is also a problem with giving the AI-based model data to train it and use it in different automated security software in the cloud ecosystem. Dataset quality and availability is still an issue for AI-based cybersecurity systems and researchers are working on it. Scalability and real-time security analysis are also very important, for AI-based software to keep web applications safe and working well.

D. Backend as the Central Processing Unit

The Backend Security Engine is the central processing unit and enables communication between all other modules [3] [6]. The backend performs user authentication, vulnerability scan initiation, AI model processing, exploitation testing, report generation, database, and API functions. The backend is responsible for performing the processing of information related to the functions described above, conducting the processing automatically and securely in the system [7] [10].

E. User Interaction

Users interact with the system via the web interface that is intended for security analysts, developers, and administrators [1] [5]. Via the web interface, users are able to perform a vulnerability scan and configure the system for security assessment, monitor the progress of scans, identify the detected vulnerabilities, and generate reports related to the security issue [8] [11].

F. Dataset Quality and Availability

The AI-based cybersecurity system needs to have a lot of different data sets to find and validate vulnerabilities and threats. This is really important. At the time researchers say that it is hard to make complete data sets for AI-based security software. This is because there is not data people are worried about privacy the databases are old and the cybersecurity environment is changing really fast. So the quality of the data that goes into the AI-based software is very important. If the data set is not correct the software will not be able to find vulnerabilities and will not work well. That is why researchers think that making data sets is a very important area of research for cybersecurity.

G. Scalability and Real-Time Security Analysis

The AI-based software should be able to handle a lot of work and support real-time security analysis. This is so that web applications can work efficiently stay safe and find threats all the time. However when there are users and services and more API calls and potential security threats it gets harder to respond quickly to threats without making the application work slower. At the time researchers are trying to make AI-based security systems better so they can process data in real-time and analyze it quickly. There is also a problem with giving the AI-based model data to train it and use it in different automated security software in the cloud ecosystem. Dataset quality and availability is still an issue for AI-based cybersecurity systems and researchers are working on it. Scalability and real-time security analysis are also very important, for AI-based software to keep web applications safe and working well.

H. Security, Privacy, and Ethical Concerns

Web security systems utilizing AI technologies involve the analysis of a significant amount of highly sensitive data, including application log files, user data, and vulnerability information, which raises several security, privacy, and ethical concerns [2] [6]. The abuse of sensitive data, vulnerabilities in AI technologies, and data breaches risk organizations using web security systems based on AI technologies [5] [9]. Moreover, the use of security recommendations provided by AI technologies and automated penetration testing should be carefully considered to avoid their misuse [8] [11]. Thus, the development of effective security measures and ethical use of AI technologies are critical for web

technologies should be based on the careful consideration of potential ethical issues and risks.

I. Integration and Interoperability Challenges

The integration of AI-based security systems with existing infrastructures is complicated by the complexity of web technologies, development platforms, cloud technologies, and security products [3,7]. Many organizations use different solutions that include a combination of static and dynamic security analysis tools, REST APIs, vulnerability scanners, and Security Information and Event Management systems [5,9]. These systems vary significantly in the data exchange formats, communication protocols, and architectures [8,11]. Therefore, the development of interoperable and extensible security solutions based on artificial intelligence is critical to ensure the effective integration of various security systems and processes [12,14].

V. RESEARCH GAPS AND FUTURE DIRECTIONS

A. Research Gaps

We found some limitations when we looked at what people have written about this topic. These limitations include:

* The problem of not being able to find zero-day vulnerabilities
* Getting a lot of positives and false negatives
* Not having AI models that can explain what they do
* Using small or old data sets
* Not being able to adapt to threats that are coming up

B. Comparative Analysis

Let us compare some of the approaches that people have talked about. We will look at:

* Machine Learning and Deep Learning to see which one is
* AI-based scanning tools and traditional scanning tools to see which one works better
* Large language models and classical Machine Learning models to see which one is more accurate
* Static analysis, which is also known as SAST and dynamic analysis, which is also known as DAST to see which one is effective
* One-agent systems and multi-agent security systems to see which one is more secure

We will look at the good and bad things about each approach how accurate they are and where they can be used.

C. Open Research Challenges

There are some challenges that still need to be solved. These include:

* Finding ways to detect zero-day attacks
* Protecting AI algorithms from attacks
* Detecting threats, in time
* Making solutions that can work for companies with a lot of IT infrastructure
* Making sure AI decision-making algorithms are transparent and explainable. That they work with existing security software

D. Future Directions

There are some directions that research could take in the future. These include:

* artificial intelligence, which is also known as XAI
* Keeping security private with federated learning
* Using AI agents on their own to do penetration testing
* Continuous learning to keep up with threats
* Using machine learning and deep learning with large language models to create something new and better.

VI. COMPARATIVE ANALYSIS AND EXPECTED OUTCOMES

A. Comparative Analysis of AI-Driven Web Security Systems

Intelligence can contribute to the security of web applications via the improvement of vulnerability detection, penetration testing, and threat analysis [1] [4].The rule-based approach has shown its effectiveness in detecting known threats; however, it has proved to be insufficient in case of more sophisticated attacks and zero-day threats [2] [6]. Meanwhile, Machine Learning and Deep Learning algorithms have demonstrated their ability to achieve better detection accuracy by learning from previous attacks, while Large Language Models and Multi-agent systems facilitate automated reasoning, exploitation, and report generation [7] [10]. At the same time, the current state of the art tools reveals several issues related to scalability, explainability, quality of data, and interoperability, which makes hybrid approaches more promising in the future [11] [14].

TABLE II: COMPARISON OF AI-DRIVEN WEB SECURITY SYSTEMS

System / Approach	Technology	Strengths	Limitations
Traditional Scanner	Rule-Based	Fast and simple	Misses advanced attacks
ML-Based Detection	Machine Learning	Better accuracy	Needs quality data
Deep Learning	Neural Networks	Detects complex threats	High computation
AI Penetration Testing	ML + AI	Automates testing	Needs regular updates
LLM-Based Security	Large Language Models	Smart analysis	May produce errors
Multi-Agent Security	AI Agents	Faster and scalable	Complex implementation

B. Expected Outcomes

The introduction of Artificial Intelligence into web applications will ensure the improvement of the accuracy, reliability, and performance of vulnerability detection and penetration testing. According to the findings of the reviewed articles, the following outcomes are expected:

- Improved Vulnerability Detection: It is expected that AI-based security software will be able to recognize both existing and potential vulnerabilities with greater precision than traditional

overlooked security vulnerabilities and improving the overall application security [2] [5].

- Enhanced Automated Penetration Testing: With the help of such technologies as Machine Learning, Large Language Models, and Multi-Agent Systems, automated penetration testing and the associated processes of vulnerability detection, exploitation verification, and assessment can be performed [6] [9].
- Reduced False Positives and False Negatives: More advanced AI is expected to allow for more accurate vulnerability classification, thus reducing the number of security alerts that are not actually vulnerable (false positives) and the number of security alerts that the software failed to identify as a threat (false negatives) [4] [8].
- Real-Time Threat Detection and Response: The ability of AI to monitor and analyze potential threats in real-time allows for immediate detection and response [7] [10].
- Intelligent Risk Assessment and Reporting: An AI-driven security framework will be able to prioritize identified vulnerabilities based on severity levels, generate a comprehensive security report, and recommend specific courses of action [11] [13].
- Scalability and Adaptability: The proposed solution is expected to provide enhanced protection for large-scale web applications, cloud infrastructure, and API-based solutions while adapting to emerging threats [12] [14].

VII. CONCLUSION

This study of the literature takes a look at how Artificial Intelligence is used to make web application security better. It does this by finding weaknesses and doing penetration testing. The study looks at research on using Machine Learning and Deep Learning to improve the security of websites. It also looks at Large Language Models and Multi-Agent Systems. The goal is to make cybersecurity solutions more accurate and efficient. The researchers looked at the parts of these systems and how they are put together. They also looked at what the systems can. Cannot do. They compared the results of using these technologies. The results show that using Artificial Intelligence to deal with cybersecurity issues is an idea. However there are some problems with using Artificial Intelligence. For example sometimes it gives results. It is also hard to understand how it works. There is not data to make it work well. It is also hard to make it work with systems. All these problems make it hard to use Artificial Intelligence for cybersecurity. So future research should focus on making Artificial Intelligence systems for cybersecurity that work well and are flexible. The future of Artificial Intelligence-driven web security systems should be the focus of this research. Artificial Intelligence-driven web security systems are the key, to making the web a safer place.

ACKNOWLEDGMENT

We are grateful to the Don Bosco Institute of Technology, Bengaluru, for giving us the desired facilities and support to successfully complete our major project. A special thanks to our Principal, Dean, Head of the Department/our Project Guide for our project. Furthermore, we express our sincerest gratitude to all our faculty members, friends and family members that have supported us and motivated us through this work.

REFERENCES

[1] I. Kertusha, G. Assres, O. Duman, and A. Arcuri, "A Survey on Web Testing: On the Rise of AI and Applications in Industry," *Science of Computer Programming*, 2026, doi: 10.1016/j.scico.2026.103473.

[2] S. S. Mahmood, "SQL Injection Detection Using Machine Learning and Explainability," *Journal of Internet Services and Information Security*, vol. 15, no. 2, pp. 309–324, 2025, doi: 10.58346/JISIS.2025.I2.022.

[3] A. Zhuravchak, A. Piskozub, B. Skorynovych, Y. Lakh, D. Zhuravchak, P. Hlushchenko, P. Venherskyi, I. Beliaiev, M. Vorokhob, and I. Kolbasynskyi, "Design and Development of a Large Language Model-Based Tool for Vulnerability Detection," *Eastern-European Journal of Enterprise Technologies*, vol. 2, no. 2 (134), pp. 75–83, 2025, doi: 10.15587/1729-4061.2025.325251.

[4] M. A. Ahmed, "An Adaptive AI-Enhanced Automated Web Security Scanner with Multi-Vector Testing and Vulnerability Chaining," 2025.

[5] Y. Stefinko, A. Piskozub, and A. Obshta, "Analysis of Vulnerability Characteristics for Automated Penetration Testing," in *Proc. IEEE Conference*, 2024, pp. 449–453.

[6] V. Saber, A. M. Bahaa-Eldin, D. ElSayad, and Z. Fayed, "Automated Penetration Testing, A Systematic Review," in *Proc. 2023 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*, 2023, pp. 373–378, doi: 10.1109/MIUCC58832.2023.10278377.

[7] P. Zhang, D. Zhang, L. Han, W. Wang, H. Wu, and Q. Zhao, "Automated Security Penetration Testing Based on Machine Learning," 2024.

[8] B. Singh, "Automating Security Testing in CI/CD Pipelines Using DevSecOps Tools: A Comprehensive Study," *Science, Technology and Development*, vol. IX, no. XII, p. 631, 2020.

[9] N. Rane and A. Qureshi, "Comparative Analysis of Automated Scanning and Manual Penetration Testing for Enhanced Cybersecurity," in *Proc. IEEE*, 2024.

[10] S. Oladele and F. Lawal, "The Impact of AI-Assisted Code Generation on Software Vulnerabilities and the Role of AI in Automated Security Testing," SSRN, 2025.

[11] S. O. Alwabisi, "AI in Penetration Testing: A Systematic Mapping Study," TechRxiv, Jun. 27, 2025, doi: 10.36227/techrxiv.175099664.46246512/v1.

[12] M. Kozlovska, A. Piskozub, and V. Khoma, "Artificial Intelligence in Penetration Testing: Leveraging AI for Advanced Vulnerability Detection and Exploitation," *Advances in Cyber-Physical Systems*, vol. 10, no. 1, pp. 65–72, 2025, doi: 10.23939/acps2025.01.065.

[13] A. Goutam and V. Tiwari, "Vulnerability Assessment and Penetration Testing to Enhance the Security of Web Application," 2024.

[14] G. Sanchez, O. Olayinka, and A. Pasikhani, "Web Application Penetration Testing with Artificial Intelligence: A Systematic Review," 2024.