Fast in Technologies and AI Solutions

Website: https://fastin-technologies.com/ftas

ISSN: xxxx-xxxx Vol. 1 No.1 Mei 2025 : 7-10.

Website Security Analysis Against SQL Injection Attack: Systematic Review

Lathiifah Faathimah[1]*, Zahyra Darrel Fauqa[2], Nabila Putri Nuraini[3], Diifta Aulya Rahmadanti[4], Nuralfisyah[5]

[1,2,3,4,5]Universitas Ahmad Dahlan

**Corresponding author E-mail: 2400018123@webmail.uad.ac.id*

Received: 21 July 2025
Accepted: 4 October 2025

Abstract

The development of information technology has driven the massive use of websites in various sectors, making it one of the main targets of cyberattacks. One of the most dangerous and frequent types of attacks is SQL Injection, which exploits loopholes in user input to execute malicious SQL commands. This study aims to analyze various methods that have been applied in efforts to secure websites against SQL Injection attacks through a systematic review approach. Seven scientific articles published between 2020 and 2025 were examined in depth by considering the methods used, the effectiveness of the protection system, and the potential security loopholes that still exist. The results of the study show that the use of Web Application Firewall (WAF), penetration testing tools such as SQLmap and OWASP ZAP, as well as input validation practices and the use of SSL/Captcha are the most effective approaches in preventing SQL Injection attacks. Nonetheless, the effectiveness of protection is highly dependent on system configuration and consistency of security implementation. Therefore, a multi-layered approach that includes secure technologies, security procedures, and software development practices is indispensable to build information systems that are resilient to SQL Injection threats.

Keywords: *SQL Injection, Website Security, Penetration Testing, Systematic Review*

1. Introduction

The rapid development of technology and information has driven digital transformation in various sectors of life, including government, education, banking and commerce. The website is one of the important components in supporting these digital activities, this is because the website is a service that is able to provide real-time information, services and transactions to users from various locations [1][2]. However, websites are the main target of cyber attacks because there are still many security holes that can be exploited by hackers[3][4]. Web application security is one of the important aspects in ensuring the integrity and sustainability of an information system[5][6]. One of the most serious threats to web application security is SQL Injection.

SQL injection is an attack technique that exploits gaps in user input by inserting SQL commands in the input form of an application which will allow the attacker to send syntax or commands to the database of an application [7](Putranto, et al., 2022). This attack is still one of the most frequently used exploitation methods and is included in the OWASP Top Ten list as the most critical web application security risk[8] [9]. Various studies have been conducted to analyze and overcome the risk of SQL injection, ranging from the implementation of prevention technologies such as Web Application Firewall (WAF), penetration testing techniques, to the development of safe programming practices [10][11]. However, a comprehensive study is still needed to evaluate and summarize the results of these studies systematically in order to provide a clearer picture of the effectiveness of website security methods against SQL Injection attacks.

Through this systematic review, it is expected to obtain an in- depth understanding of the methods that have been applied to protect websites from SQL Injection attacks, the level of effectiveness, as well as practical recommendations for the development and security practitioners in strengthening information systems.

2. Literature review

Based on the review of seven scientific articles that have been reviewed in this research, various approaches were found to be applied in securing websites against SQL Injection attacks. The most commonly used methods include the implementation of Web Application Firewall (WAF), penetration testing techniques with the help of tools such as SQLmap and OWASP ZAP, as well as additional security measures such as input validation, the use of SSL/TLS protocols, and Captcha. Studies by WAF implementation proved effective in

blocking SQL Injection attacks before the malicious commands reach the database system. These findings indicate that WAF serves as a very important first layer of defense in web application security systems. Meanwhile, the penetration testing method is the main tool in detecting security holes that are not detected by passive security systems. Tested several sites using SQLmap and OWASP ZAP, the results varied, ranging from sites that were able to withstand attacks to those that were vulnerable due to weak input validation and the absence of additional protection systems. This emphasizes the importance of regular security testing so that potential vulnerabilities can be detected and addressed before they are misused by irresponsible parties.

In addition, research by [12][13] also underscores the importance of additional measures such as the implementation of SSL and Captcha. SSL serves to encrypt the communication between the user and the server, while Captcha limits automated activities that are commonly used in attack scenarios. Thecombination of these elements can improve the security of both the backend and frontend of the application. Overall, this literature review confirms that the best approach to counteracting SQL Injection attacks is through a defense in depth strategy that combines protection technologies, secure software development practices, and active security testing. Protection cannot rely on just one method, but must be designed thoroughly and consistently so that information systems can defend against increasingly complex cyber threats.

3. Research Method

This research uses a systematic review method to collect, analyze, and synthesize research results related to “Website Security against SQL Injection Attacks”. This method was chosen to provide a comprehensive and structured overview of the various approaches and techniques that have been used in overcoming the threat of SQL Injection in web applications. The following are the stages of systematic review research:

3.1. Research question formulation

This research focuses on the main question, namely “what are the methods and technologies that have been applied to secure websites from SQL Injection attacks and how effective are these methods?”.

3.2. Literature search

The lieterature search in this research was conducted by accessing several academic databases, namely Google Scholar and Pubmed Article. The keywords used include “SQL Injection”, “website security”, “web application firewall”, and “SQL Injection mitigation”. The search focused on articles and journals published in the last 5 years, namely 2020 - 2025.

3.3. Inclusion and exclusion criteria

The inclusion and exclusion criteria for articles and journals that can be used in this research are that the articles should focus on analyzing website security against SQL Injection attacks. In addition, the articles must use security testing methods such as penetration testing, tools such as SQLmap and OWASP ZAP, or the implementation of a web application firewall (WAF) as an effort to protect against such attacks.

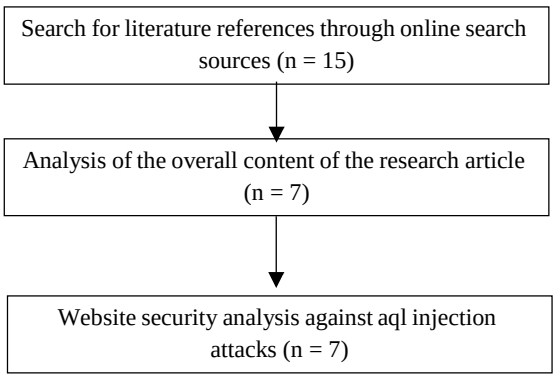
3.4. Study selection

The selection process was conducted in two stages. The first stage was an initial screening based on the title and abstract to exclude irrelevant articles. The second stage was a full-text review to ensure that the articles met the inclusion criteria.

3.5. Data extraction

The important data taken from each article are the author's name, year of publication, research title, methods used, and the main results or findings that match the research questions.

Table 1: Diagram of literature search



The important data taken from each article are the author's name, year of publication, research title, methods used, and the main results or findings that match the research questions. Of the 15 articles, 7 articles were obtained that discussed the security of websites against SQL injection attacks. The 7 articles met the inclusion and exclusion criteria and fit the research question.

4. Results and Discussion

The results found from the seven articles are that the main methods and technologies used to protect websites from SQL Injection attacks consist of the use of Web Application Firewall (WAF), penetration testing tools, and the implementation of additional security measures such as input validation, SSL, and Captcha. Two studies, namely by [14] and [15], show that the use of WAF is significantly able to block SQL Injection attacks before the malicious query reaches the database. In both studies, the tested systems proved to be able to resist attacks thanks to the presence of a good WAF configuration, making it one of the effective protection technologies against SQL- based attacks.

In addition, penetration testing methods are also widely used in the reviewed studies, as seen in the studies by [16]. Tools such as SQLmap and OWASP ZAP were used to simulate attacks and identify security holes. The results showed variations: sites such as District Court X successfully resisted SQL Injection attacks due to strong firewall protection, while other sites such as Shih Ka Plastic Boxes Factory and SMK Wongsorejo showed high vulnerability due to weak input validation. These findings demonstrate the importance of implementing periodic security testing to anticipate exploitation of loopholes undetected by passive protection systems.

Meanwhile, a study by [17] showed that additional security measures such as the use of Secure Socket Layer (SSL) and Captcha codes, as well as input validation practices on the user side, also provide additional protection against SQL Injection attacks. This approach strengthens the security of both the application development side and the user interaction with the system.

Overall, the review showed that the effectiveness of SQL Injection attack mitigation is highly dependent on a combination of protection technologies, secure development practices, and active security testing. Web Application Firewall is the first line of defense which proved to be effective, but without proper input validation and regular security testing, the system remains at risk of exploitation. Penetration testing using tools such as SQLmap and OWASP ZAP is proven to identify security holes with high precision, while development practices such as SSL and Captcha implementation can strengthen protection on the frontend. Therefore, effectively securing a website from SQL Injection threats requires a layered and comprehensive approach, rather than relying on just one method.

5. Conclusion

Based on the results of a systematic review of seven studies that discuss website security against SQL Injection attacks, it can be concluded that SQL Injection is still one of the serious threats to web applications, especially if not accompanied by adequate security mechanisms. Some technologies such as Web Application Firewall (WAF) are proven to be effective in blocking attacks before they reach the database system. In addition, the use of penetration testing tools such as SQLmap, OWASP ZAP, and WPScan proved useful in detecting security holes that are not identified by passive protection systems.

However, the effectiveness of protection against SQL Injection depends not only on the use of security tools, but also on secure software development practices, such as validation and sanitization of user input, use of SSL, and Captcha codes. The studies analyzed show that the best approach to preventing SQL Injection is through a combination of protection technologies, regular testing, and implementation of a comprehensive web application security policy.

References

[1] C. L. Martin, E. N. Kramer-Kostecka, J. A. Linde, S. Friend, V. R. Zuroski, and J. A. Fulkerson, “Leveraging interdisciplinary teams to develop and implement secure websites for behavioral research: Applied tutorial,” *J Med Internet Res*, vol. 22, no. 9, p. e19217, Sep. 2020, doi: 10.2196/19217.

[2] P. Makris *et al.*, “Digitization era for electric utilities: A novel business model through an inter-disciplinary s/w platform and open research challenges,” *IEEE Access*, vol. 6, pp. 22452–22463, Apr. 2018, doi: 10.1109/ACCESS.2018.2828323.

[3] B. Fischer, D. Meissner, R. Nyuur, and D. Sarpong, "Guest Editorial: Cyber-Attacks, Strategic Cyber-Foresight, and Security," *IEEE Trans Eng Manag*, vol. 69, no. 6, pp. 3660–3663, Dec. 2022, doi: 10.1109/TEM.2022.3204165.

[4] P. Papadopoulos, P. Ilia, M. Polychronakis, E. P. Markatos, S. Ioannidis, and G. Vasiliadis, "Master of Web Puppets: Abusing Web Browsers for Persistent and Stealthy Computation", doi: 10.14722/ndss.2019.23070.

[5] W. B. Demilie and F. G. Deriba, "Detection and prevention of SQLI attacks and developing compressive framework using machine learning and hybrid techniques," *J Big Data*, vol. 9, no. 1, pp. 1–30, Dec. 2022, doi: 10.1186/S40537-022-00678-0/FIGURES/7.

[6] B. R. Dawadi, B. Adhikari, and D. K. Srivastava, "Deep Learning Technique-Enabled Web Application Firewall for the Detection of Web Attacks," *Sensors 2023, Vol. 23, Page 2073*, vol. 23, no. 4, p. 2073, Feb. 2023, doi: 10.3390/S23042073.

[7] W. B. Demilie and F. G. Deriba, "Detection and prevention of SQLI attacks and developing compressive framework using machine learning and hybrid techniques," *J Big Data*, vol. 9, no. 1, pp. 1–30, Dec. 2022, doi: 10.1186/S40537-022-00678-0/FIGURES/7.

[8] F. M. M. Mokbal, W. Dan, A. Imran, L. Jiuchuan, F. Akhtar, and W. Xiaoxi, "MLPXSS: An Integrated XSS-Based Attack Detection Scheme in Web Applications Using Multilayer Perceptron Technique," *IEEE Access*, vol. 7, pp. 100567–100580, 2019, doi: 10.1109/ACCESS.2019.2927417.

[9] M. Liu, B. Zhang, W. Chen, and X. Zhang, "A Survey of Exploitation and Detection Methods of XSS Vulnerabilities," *IEEE Access*, vol. 7, pp. 182004–182016, 2019, doi: 10.1109/ACCESS.2019.2960449.

[10] S. Abaimov and G. Bianchi, "CODDLE: Code-Injection Detection with Deep Learning," *IEEE Access*, vol. 7, pp. 128617–128627, 2019, doi: 10.1109/ACCESS.2019.2939870.

[11] T. S. Riera, J. R. B. Higuera, J. B. Higuera, J. J. M. Herraiz, and J. A. S. Montalvo, "Prevention and Fighting against Web Attacks through Anomaly Detection Technology. A Systematic Review," *Sustainability 2020, Vol. 12, Page 4945*, vol. 12, no. 12, p. 4945, Jun. 2020, doi: 10.3390/SU12124945.

[12] D. Hitaj, B. Hitaj, S. Jajodia, and L. V. Mancini, "Capture the Bot: Using Adversarial Examples to Improve CAPTCHA Robustness to Bot Attacks," *IEEE Intell Syst*, vol. 36, no. 5, pp. 104–112, 2021, doi: 10.1109/MIS.2020.3036156.

[13] Z. A. Alizai, H. Tahir, M. H. Murtaza, S. Tahir, and K. McDonald-Maier, "Key-Based Cookie-Less Session Management Framework for Application Layer Security," *IEEE Access*, vol. 7, pp. 128544–128554, 2019, doi: 10.1109/ACCESS.2019.2940331.

[14] B. R. Dawadi, B. Adhikari, and D. K. Srivastava, "Deep Learning Technique-Enabled Web Application Firewall for the Detection of Web Attacks," *Sensors 2023, Vol. 23, Page 2073*, vol. 23, no. 4, p. 2073, Feb. 2023, doi: 10.3390/S23042073.

[15] W. B. Demilie and F. G. Deriba, "Detection and prevention of SQLI attacks and developing compressive framework using machine learning and hybrid techniques," *J Big Data*, vol. 9, no. 1, pp. 1–30, Dec. 2022, doi: 10.1186/S40537-022-00678-0/FIGURES/7.

[16] F. Faisal Fadlalla and H. T. Elshoush, "Input Validation Vulnerabilities in Web Applications: Systematic Review, Classification, and Analysis of the Current State-of-the-Art," *IEEE Access*, vol. 11, pp. 40128–40161, 2023, doi: 10.1109/ACCESS.2023.3266385.

[17] W. B. Demilie and F. G. Deriba, "Detection and prevention of SQLI attacks and developing compressive framework using machine learning and hybrid techniques," *J Big Data*, vol. 9, no. 1, pp. 1–30, Dec. 2022, doi: 10.1186/S40537-022-00678-0/FIGURES/7.