IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

A Survey on SQL Injection Detection Techniques for Website Security

[1]Pooja panadiya, [2]Prof. Manish Kumar Singhal
[1]*M.tech Scholar*, [2]*Associate Professor & H.O.D*
[1,2]*Department of Information Technology (IT)*
[1,2] *NRI Institute Of Information Science And Technology, Bhopal (Mp), India,*
[1]ppanadiya5@gmail.com [2]*manishsinghal.nirt@gmail.com*

Abstract : In this survey paper discuss the SQL Injection (SQLi) is a critical web security vulnerability that occurs when attackers exploit input fields on websites to inject malicious SQL queries. These attacks can lead to unauthorized data access, manipulation, or deletion, posing severe threats to the confidentiality and integrity of sensitive information. Detecting SQL injections involves various techniques aimed at preventing such breaches. Signature-based detection identifies known attack patterns, while anomaly-based methods flag deviations from typical behavior in SQL queries. This survey paper explores the Additionally, input validation techniques like whitelisting and parameterized queries help prevent harmful SQL code execution. These methods, when combined with web application firewalls (WAFs) and security testing tools, form a robust defense mechanism to safeguard websites against SQL injection attacks, ensuring higher levels of security and reliability for web applications.

Keyword:- **— SQL injection attack, SQLIA prevention, Query Transformation, Normalization of Queries, Document Similarity, Hidden Markov Model, Support Vector Machine, Graph of Tokens, Centrality Measures, Feature Selection, etc.**

INTRODUCTION

In this Internet age, web-based applications have become an integral part of our daily lives. The Internet and web applications are the modern day workplace and business ground contributing to drive the world economy. At the same time, hackers and attackers have developed a parallel underground economy of hacking into web applications and stealing large amount of sensitive business-critical information with malicious intent. Incidents of data-breach have become frequent causing huge financial losses to organizations and serious impact for the users in various levels. Therefore, securing web applications from the prying eyes of attackers has become a primary concern. Among the various security threats a web application is exposed to, SQL Injection Attack (SQLIA) has taken the forefront. It has prevailed as a popular attack method since last several years. Using SQL injection attack, an attacker can extract, modify, or destroy the backend database of web applications. The simplicity of attacking a web application using SQL injection and abundance of vulnerable applications on the Internet has largely contributed to widespread data breach incidents. Discovered in late 1998, the problem still persists with almost the same vigor and intensity. It has continued to evolve with new forms and features, posing new challenges for the research community.

SQL injection is one of the most common and dangerous vulnerabilities that affect websites and web applications. It occurs when attackers manipulate user inputs to execute malicious SQL queries, potentially gaining unauthorized access to databases, extracting sensitive information, or compromising the integrity of data. This form of attack exploits weak input validation, allowing malicious users to inject code directly into the backend database queries. SQL injection poses a serious threat to website security, particularly for applications handling sensitive data such as personal information, financial records, or login credentials.

To mitigate this threat, various SQL injection detection techniques have been developed, each offering different levels of effectiveness based on the complexity and structure of the website. Detection methods can be broadly categorized into signature-based, anomaly-based, and hybrid techniques. Signature-based detection relies on identifying known patterns of SQL injection attacks, while anomaly-based techniques focus on detecting deviations from normal behavior in database interactions. Hybrid approaches combine both strategies for enhanced detection accuracy.

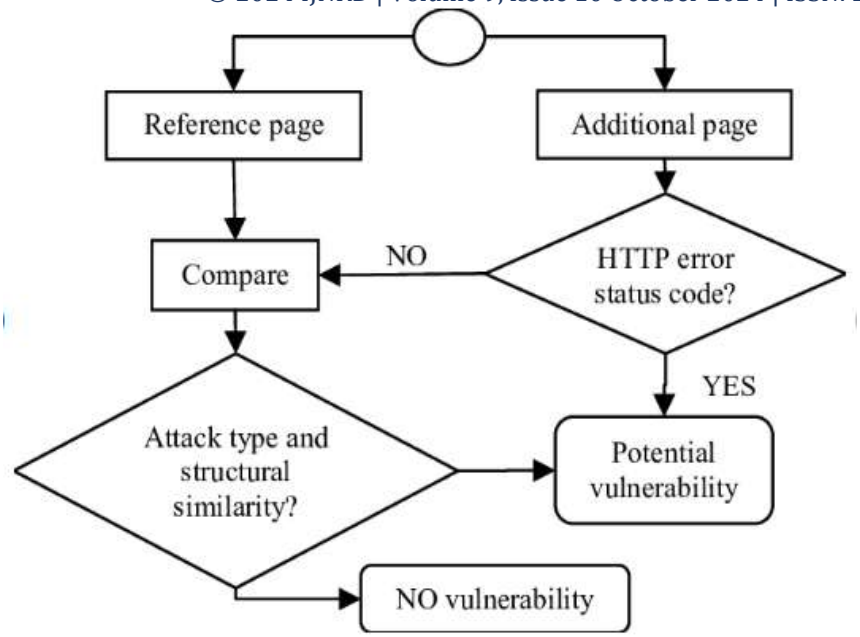

Fig 1 Detection of SQL Injection

Developers must adopt secure coding practices, such as using parameterized queries, implementing proper input validation, and employing web application firewalls (WAFs) to filter and monitor traffic. By employing robust SQL injection detection techniques and maintaining security best practices, websites can significantly reduce the risk of attack and safeguard their data from malicious actors.

LITERATURE SURVEY

Abdulrasheed Jimoh, et.al (2024), Author are presented a This research offers insightful information about various algorism for the SQLI attacks detection within web-based applications. The results highlight the significance of choosing the right model to fortify web application security, with CNN, and GRU emerging as strong contenders. As the digital landscape continues to evolve, understanding the strengths and weaknesses of these models is vital for ensuring the robustness of web applications against SQLI attacks. To effectively neutralize SQLi attack, the adoption of advanced techniques Capsule Networks, Transformer-based Models is strongly advised in future while a larger dataset not only enhances the model's ability to detect and classify SQLI attacks but also improves its generalization capabilities. It allows the model to learn intricate patterns, variations, and anomalies associated with SQLI attacks, thereby boosting its accuracy and reliability in realworld scenarios [01].

Bahman Arasteh,et.al,(2024), Author are analysis SQL injection is one of the important security issues in web applications because it allows an attacker to interact with the application's database. SQL injection attacks can be detected using machine learning algorithms. The effective features should be employed in the training stage to develop an optimal classifier with optimal accuracy. Identifying the most effective features is an NP-complete combinatorial optimization problem. Feature selection is the process of selecting the training dataset's smallest and most effective features. The main objective of this study is to enhance the accuracy, precision, and sensitivity of the SQLi detection method. In this study, an effective method to detect SQL injection attacks has been proposed. In the first stage, a specific training dataset consisting of 13 features was prepared. In the second stage, two different binary versions of the Gray-Wolf algorithm were developed to select the most effective features of the dataset. The created optimal datasets were used by different machine learning algorithms. Creating a new SQLi training dataset with 13 numeric features, developing two different binary versions of the gray wolf optimizer to optimally select the features of the dataset, and creating an effective and efficient classifier to detect SQLi attacks are the main contributions of this study [02]

Mohammed A M Oudah, et.al, (2024), Author are study a new SQL injection attacks are critical security vulnerability exploitation in web applications, posing risks to data, if successfully executed, allowing attackers to gain unauthorised access to sensitive data. Due to the absence of a standardised structure, traditional signature-based detection methods face challenges in effectively detecting SQL injection attacks. To overcome this challenge, machine learning (ML) algorithms have emerged as a promising approach for detecting SQL injection attacks. This paper presents a comprehensive literature review on the utilisation of ML techniques for SQL injection detection. The review covers various aspects, including dataset collection, feature extraction, training, and testing, with different ML algorithms. The studies included in the review demonstrate high levels of accuracy in detecting attacks and reducing false positives [03].

Fawaz Khaled Alarfaj et.al, (2023), Researcher are Comparative analysis is presented here of SQL injection attack is considered one of the most dangerous vulnerabilities exploited to leak sensitive information, gain unauthorized access, and cause financial loss to individuals and organizations. Conventional defense approaches use static and heuristic methods to detect previously known SQL injection attacks. Existing research uses machine learning techniques that have the capability of detecting previously unknown and novel attack types. Taking advantage of deep learning to improve detection accuracy, we propose using a probabilistic neural network (PNN) to detect SQL injection attacks. To achieve the best value in selecting a smoothing parament, we employed the BAT algorithm, a metaheuristic algorithm for optimization. In this study, a dataset consisting of 6000 SQL injections and 3500 normal queries was used. Features were extracted based on tokenizing and a regular expression and were selected using Chi-Square testing. The features used in this study were collected from the network traffic and SQL queries. The

experiment results show that our proposed PNN achieved an accuracy of 99.19% with a precision of 0.995%, a recall of 0.981%, and an F-Measure of 0.928% when employing a 10-fold cross-validation compared to other classifiers in different scenarios [04].

Nanang Cahyadi, et. al (2023), Authors presented SQL injection attacks (SQLIAs) pose increasing threats as more organizations adopt vulnerable web applications and databases. By manipulating queries, SQLIAs access and destroy confidential data. This paper delivers three contributions around improving SQLIA detection research: first, a literature review assessing current detection/prevention systems to produce an SQL injection detection framework; second, specialized deep learning models optimizing session pattern analysis and feature engineering to enhance performance; third, comparing proposed models against previous defenses to surface promising research directions. Results highlight opportunities like real-time systems generalizing across attack variants through emerging techniques. Additionally, with attack complexity rising, systematized SQLIA investigation is warranted. Despite extensive study, current perspectives lack cohesive guidance informing mitigation strategies. Therefore, a framework is proposed holistically mapping knowledge gaps around contemporary SQLIAs, seminal threats in web applications, and security solutions. Furthermore, a multi-faceted framework examines research trends divided into hardening existing apps, detecting attacks on production systems, and integrating secure development practices. Literature suggests comprehensive resilience requires concurrent strength across these areas. Finally, future work remains in integrated frameworks, deep reinforcement learning adoption, automated AI auditing, and differential privacy to advance real-world SQL injection detection and prevention [05].

Taseer Muhammad et.al (2022) - The model's accuracy, true-positive rate, false-positive rate, and time to develop the model all performed well, according to the findings of the performance assessment of the model for the detection and categorization of the SQLIA. The machine learning paradigm may be used to construct pattern matching, which has the ability to mitigate SQLIA queries made via login, URL, and search [27]. This study effectively identified malicious log files by using machine learning to distinguish between malicious and benign online requests produced from access log files. Additionally, string matching is used during the categorization step to match the characteristics. The primary challenge for SQLI research is finding reliable and appropriate internet datasets. Therefore, data gathering is created internally by establishing a straightforward login page and carrying out SQL Injection assaults. Fortunately, platforms like DVWA exist that can be used to produce datasets by doing injections. Only a small number of the SQL injection dataset's samples may thus be utilized for training and testing [6].

Yazeed Abdulmalik et.al (2021) - SQL Injection Attack (SQLIA) is a common cyberattack that target web application database. With the ever increasing and varying techniques to exploit web application SQLIA vulnerabilities, there is no a comprehensive method that can solve this kind of attacks. Therefore, these various of attack techniques required to establish many methods against in order to mitigate its threats. However, most of these methods have not yet been evaluated, where it is still just theories and require to implement and measure its performance and set its limitation. Moreover, most of the existing SQL injection countermeasures either used syntax-based detection methods or a list of predefined rules to detect the SQL injection, which is vulnerable in advance and sophisticated type of attacks because attackers create new ways to evade the detection utilizing their pre-knowledge. Although semantic-based features can improve the detection, up to our knowledge, no studies focused on extracting the semantic features from SQL stamens. This paper, investigates a designed model that can improve the efficacy of the SQL injection attack detection using machine learning techniques by extracting the semantic features that can effectively indicate the SQL injection attack [7]

SQL INJECTION ATTACK

SQL Injection (SQLi) is a type of cyber attack that targets the database layer of web applications. It occurs when an attacker manipulates a vulnerable input field to inject malicious SQL code into a query that the application sends to its database. This allows the attacker to bypass authentication, retrieve, modify, or delete sensitive data, and even execute administrative operations on the database. SQL Injection exploits flaws in input validation, where the application does not properly sanitize user inputs. As a result, attackers can manipulate queries to extract private information such as login credentials, personal data, or financial records. This type of attack is particularly dangerous because it can affect any website or application that relies on SQL databases, posing serious security risks to organizations by exposing their data to unauthorized access. Prevention measures include using prepared statements, input validation, and limiting database permissions.

In addition to compromising data, SQL Injection attacks can also allow attackers to take control of the underlying server, escalate privileges, or install malicious software, making it a multifaceted threat. For example, an attacker could gain administrative access to a system and manipulate the entire database or access other systems connected to it. Some advanced forms of SQLi, like blind SQL Injection, do not directly return results to the attacker but still enable them to deduce valuable information through careful analysis of responses.

The consequences of a successful SQL Injection can be severe, including data breaches, loss of trust, legal liabilities, and financial damage. Many high-profile data breaches in the past have involved SQL Injection attacks, showcasing the widespread impact such vulnerabilities can have. Organizations handling sensitive customer information, like financial institutions, e-commerce platforms, and government agencies, are prime targets.

SQL Injection Attack (SQLIA) is a subset of the unverified/unsanitized input vulnerability and occurs when an attacker attempts to change the logic, semantics or syntax, and behavior of a legitimate Structured Query Language (SQL) statement by inserting additional SQL keywords and/or operators into the statement through the URL query string parameters or form fields, usually with a malicious intent. By injecting specially crafted inputs with SQL keywords, delimiters, operators, etc., the attacker attempts to change the results of the SQL query, thereby altering the HTML content returned by the server. Using this injection technique, the attacker can gain unauthorized access to restricted areas of a web application, and also be able to retrieve, alter, or damage the information in the backend database. In most cases, the intention of the attacker is to steal the sensitive information, such as credit card details, email addresses, passwords, and other private information, etc., stored in the backend database

CONCLUSION

In this survey paper discuss on SQL injection detection techniques are critical for maintaining website security, as they help to identify and mitigate one of the most prevalent cyber threats. By implementing robust detection mechanisms such as signature-based, anomaly-based, and machine learning approaches, organizations can safeguard sensitive data and prevent unauthorized access. While traditional methods like input validation and parameterized queries form a solid defense, advanced techniques such as machine learning enhance the ability to detect complex and evolving SQL injection attacks. Therefore, a multi-layered security strategy combining proactive detection, real-time monitoring, and regular updates is essential for effective protection against SQL injection vulnerabilities.

REFERENCES

[1] Abdulrasheed Jimoh, Muhammed Kabir Ahmed, Suraj Salihu, Bala Mod, and Mohammed Nasir Salihu. " Enhancing Web Security Through Comprehensive Evaluation of SQL Injection Detection Models." 23–25 May 2024.

[2] Bahman Arasteh1, Babak Aghaei, Behnoud Farzad Keyvan Arasteh1 Farzad Kiani4 Mahsa Torkamanian-Afshar. " Detecting SQL injection attacks by binary gray wolf optimizer and machine learning algorithms." Volume 36, pages 6771–6792, (2024) 27 February 2024.

[3] Mohammed A M Oudah and Mohd Fadzli Marhusin. " SQL Injection Detection using Machine Learning." Volume 10, Issue No. 1 eISSN: 2601-0003, 5 April 2024.

[4] Fawaz Khaled Alarfaj and Nayeem Ahmad Khan. "Enhancing the Performance of SQL Injection Attack Detection through Probabilistic Neural Networks." Volume 13, Issue 7, 29 March 2023.

[5] Nanang Cahyadi, Syifa Nurgaida Yutia , Pietra Dorand. "Enhancing SQL Injection Protection Through Integration, Automation, and Privacy." 19-12-2023.

[6] Taseer Muhammad, Hamayoon Ghafory "SQL Injection Attack Detection Using Machine Learning Algorithm," Vol.2022, pp. 5–17, 25 Feb 2022.

[7] I Yazeed Abdulmalik, "An Improved SQL Injection AttackDetectionModelUsing Machine Learning Techniques" 24/05/2021, DOI: https://doi.org/ 10.11113/ijic.v11n1.300

[8] S. Srivastava, "A Survey On: Attacks due to SQL injection and their prevention method for web application."

[9] G. Goos et al., "Structured Object-Oriented Formal Language and Method." [Online]. Available: http://www.springer.com/series/7407

[10] M. Leithner, B. Garn, and D. E. Simos, "HYDRA: Feedback-driven black-box exploitation of injection vulnerabilities," Inf Softw Technol, vol. 140, Dec. 2021, doi: 10.1016/j.infsof.2021.106703.

[11] 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.

[12] X. Xie, C. Ren, Y. Fu, J. Xu, and J. Guo, "SQL Injection Detection for Web Applications Based on Elastic-Pooling CNN," IEEE Access, vol. 7, pp. 151475–151481, 2019, doi: 10.1109/ACCESS.2019.2947527.

[13] W. Zhang et al., "Deep Neural Network-Based SQL Injection Detection Method," Security and Communication Networks, vol. 2022, 2022, doi: 10.1155/2022/4836289.

[14] Q. Li, F. Wang, J. Wang, and W. Li, "LSTM-Based SQL Injection Detection Method for Intelligent Transportation System," IEEE Trans Veh Technol, vol. 68, no. 5, pp. 4182–4191, May 2019, doi: 10.1109/TVT.2019.2893675.

[15] N. Gandhi, J. Patel, R. Sisodiya, N. Doshi, and S. Mishra, "A CNN-BiLSTM based Approach for Detection of SQL Injection Attacks," in Proceedings of 2nd IEEE International Conference on Computational Intelligence and Knowledge Economy, ICCIKE 2021, Institute of Electrical and Electronics Engineers Inc., Mar. 2021, pp. 378–383. doi: 10.1109/ICCIKE51210.2021.9410675.

[16] Q. Li, W. Li, J. Wang, and M. Cheng, "A SQL Injection Detection Method Based on Adaptive Deep Forest," IEEE Access, vol. 7, pp. 145385–145394, 2019, doi: 10.1109/ACCESS.2019.2944951.

[17] R. K. Dhanaraj et al., "Random Forest Bagging and X-Means Clustered Antipattern Detection from SQL Query Log for Accessing Secure Mobile Data," Wirel Commun Mob Comput, vol. 2021, 2021, doi: 10.1155/2021/2730246.

[18] L. Zhang, D. Zhang, C. Wang, J. Zhao, and Z. Zhang, "ART4SQLi: The ART of SQL Injection Vulnerability Discovery," IEEE Trans Reliab, vol. 68, no. 4, pp. 1470–1489, Dec. 2019, doi: 10.1109/TR.2019.2910285.

[19] M. Ahmed et al., "PhishCatcher: Client-Side Defense Against Web Spoofing Attacks Using Machine Learning," IEEE Access, vol. 11, pp. 61249–61263, 2023, doi: 10.1109/ACCESS.2023.3287226.

[20] R. Sreejith and S. Senthil, "Dynamic Data Infrastructure Security for Interoperable e-Healthcare Systems: A Semantic FeatureDriven NoSQL Intrusion Attack Detection Model," Biomed Res Int, vol. 2022, 2022, doi: 10.1155/2022/4080199.

[21] C. Taylor and S. Sakharkar, "DROP TABLE textbooks: An argument for SQL injection coverage in database textbooks," in SIGCSE 2019 - Proceedings of the 50th ACM Technical Symposium on Computer Science Education, Association for Computing Machinery, Inc, Feb. 2019, pp. 191–197. doi: 10.1145/3287324.3287429.

[22] B. Aruna and B. Usharani, "SQLID Framework in Order ToPerceive SQL Injection Attack on Web Application," in IOP Conference Series: Materials Science and Engineering, IOP Publishing Ltd, 2020. doi: 10.1088/1757-899X/981/2/022013.

[23] S. Ibarra-Fiallos, J. B. Higuera, M. Intriago-Pazmino, J. R. B. Higuera, J. A. S. Montalvo, and J. Cubo, "Effective Filter for Common Injection Attacks in Online Web Applications," IEEE Access, vol. 9, pp. 10378–10391, 2021, doi: 10.1109/ACCESS.2021.3050566.

.