Int. J. Nonlinear Anal. Appl. 13 (2022) 1, 3773-3782
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2022.6152

Review of SQL injection attacks: Detection, to enhance the security of the website from client-side attacks

Manar Hasan Ali AL-Maliki[a,*], Mahdi Nsaif Jasim[a]

[a] *Computer science Department, Informatics Institute for Postgraduate Studies, Iraq*
[b] *University of Information Technology and Communications, Iraq*

(Communicated by Javad Vahidi)

Abstract

The importance of cyber-security in protecting data and information is huge in this era of technology. With the number of cyber-attacks increasing daily, the security system has been developing for several years because we have been concerned about predicting and preventing cyber-attacks. The top 10 security threats identified by the OWASP are injection attacks. The most common vulnerability is SQL injection and is the most dangerous security vulnerability due to the multiplicity of its types and the rapid changes that can be caused by SQL injection and may lead to financial loss, data leakage, and significant damage to the database and this causes the site to be paralyzed. Detecting SQL injections is still a difficult task. How to successfully defend against SQL injection attacks has become the focus and frontier of web security in recent years. Machine learning has proven successful against these threats and effectively prevents and detects cross-site scripting and SQL input in web applications. Machine learning is used to analyze and identify security vulnerabilities. It uses classic machine learning algorithms and deep learning to evaluate the classified model using input validation features.

Keywords: Machine Learning, SQL Injection; Neural Network, Deep Learning Introduction

1. Introduction

SQL Injection is one of the most well-known and powerful web application attacks and the most dangerous one. Show a snapshot of the essential ICT indicators, including current-year projections according to the most recent figures, 5.2 billion people, or roughly 63 percent of the world's population, will be using the Internet in 2021 [19]. This represents a nearly 17 percent increase over the previous years, with nearly 800 million people estimated to have gone online during that time. According to the Open WEB Application Security Project (OWASP), SQL injection attacks are also the most dangerous to web-based programs and ranked third among the threats in 2021 [17]. It contains known flaws, including cross-site scripting and SQL injection. With this massive growth in Internet users, a lot of time and resources are being spent to make the Internet a safer place, but there are constantly increasing threats to user privacy every day. the Internet cannot be dispensed with because it has become an essential part of the modern environment.The greatest solution, however, is to create the Internet a secure environment in which users can share sensitive information. There are a number of SQL Injection detection tools available that can reveal any SQL Injection vulnerability in source code that has gone unnoticed. The structured query language (SQL) is a programming language for working with relational databases, which are considered the backbone of the web-based

[*] Corresponding author
Email addresses: manarh637@gmail.com (Manar Hasan Ali AL-Maliki), mahdinsaif@uoitc.edu.iq (Mahdi Nsaif Jasim)

Received: June 2021 *Accepted:* October 2021

application. They are controlled by Structured Query Language statements (SQL). Different SQL statements are used to carry out the interaction. SQL injection (SQLI) is an attack that takes advantage of SQL statement inputs. SQL queries are generally contaminated with special characters or keywords to carry out the attacks. The attacker attempts to change the statement's logic to read secret database records, modify them, and delete them. SQL injection attacks have far-reaching consequences, which differ depending on the database applications. Authentication bypass, information exposure, compromised data integrity, compromised data availability, and remote command execution are just a few of the consequences of successful SQL injection attacks. Authentication bypass allows attackers to gain access to a database application by using a fictitious username and password, information disclosure allows them to obtain sensitive data from the database, compromised data integrity allows them to change the contents of the database, compromised availability data allows them to delete data from the database, and remote command execution allows them to affect the host operating system [1]. The primary goal of this study is to evaluate recent research on machine learning algorithms for detecting SQL-Injection in web applications. A survey review of the literature is done to accomplish this. The following are the precise research questions for this review:

- What is the study's goal?
- What is the Machine Learning approach for detecting SQL Injection?
- What are the different types of SQL Injection?

This publication aims to contribute to the field of cyberattacks research by offering a set of data and domain knowledge that another researcher may use to broaden the study or construct a cyber-attack detection project.

2. Machine Learning

The notion of machine learning, abbreviated as ML, can be summarized as one of the branches arising from the study of artificial intelligence (AL) focused on programming computers of various types to do tasks and carry out commands handed to them depending on and analyzing data. It's worth noting that the term ”machine learning” first appeared in 1959 at the instruction of Arthur Samuel, the father of artificial intelligence, within the scope of IBM laboratories' work. It's also worth noting that the machine, in this case, must rely on the analysis of pre-entered data to meet the commands and tasks required of it; machine learning has two different types: supervised and unsupervised. Machine Learning Techniques Supervised and unsupervised learning are two of the most extensively used machine learning approaches. The majority of machine learning (about 70%) involves supervised learning. Unsupervised learning accounts for ten to twenty percent of all learning. Other occasionally utilized technologies include semi-supervised and reinforcement learning [2].

3. What is SQL injection?

SQL injection, often known as SQLI, is a frequent attack vector in which malicious SQL code is used to modify backend databases and get access to data that was not intended to be revealed. This data could contain anything from sensitive corporate data to user lists to private consumer information. Unauthorized access of user lists and the destruction of entire tables are all possible outcomes of a successful attack and, in certain cases, the attacker gains administrative rights to a database and can stole the user's personal information such as phone numbers, addresses, and credit card details all of which are highly detrimental to a business and in that case the company loses the customer trust [18].

Types of SQL Injection.

A. Impair Input: If a client-side script only manages an input at the application's front end, cross-site scripting can be used to circumvent the security function. As a result, the attacker can submit malicious data into the server without being validated.

B. Client-side regulation: If a client-side script only regulates an input at the application's front-end, cross-site scripting can be used to circumvent the security feature. As a result, the attacker can submit malicious data into the server without being validated.

C. Error message displayed: An SQL code error message is displayed on the web. A website programmer's error or a setup issue are the most common causes of this error. Attackers can utilize exposed information, such as table names or database names, to perform SQL Injection.

D. Uncontrolled variable: A database variable that isn't constrained by a script allows an attacker to edit the data using a SQL statement.

E. Tautologies: Using the conditional OR operator, the code is injected into a query's WHERE conditional, ensuring that the query continuously evaluates to TRUE. The query evaluates to true for each row in the table since the conditional is a tautology.

The SQL injection issue impacts not only the security of each website's data, but can also pose a threat to a server's or network's entire database system. This can result in a website crash, data alterations, privacy leakage, sensitive information disclosure, malware proliferation, and network paralysis [3].

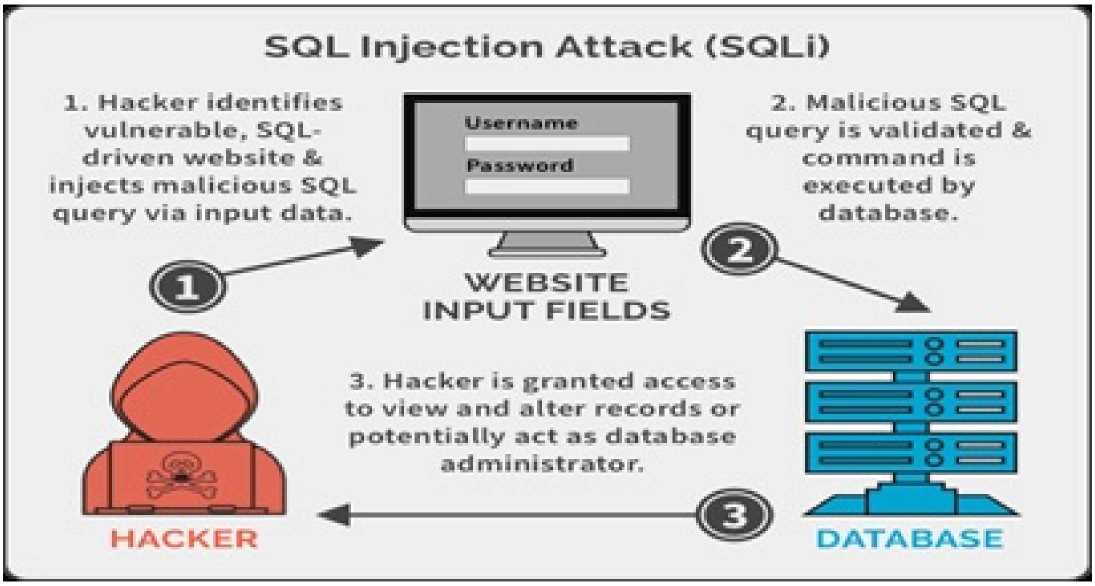
SQL Injection Attack (SQLi)
1. Hacker identifies vulnerable, SQL-driven website & injects malicious SQL query via input data.
Username
Password
WEBSITE INPUT FIELDS
2. Malicious SQL query is validated & command is executed by database.
1
2
3
3. Hacker is granted access to view and alter records or potentially act as database administrator.
HACKER
DATABASE

Figure 1: SQL injection attack web-based application security [16].

4. Methods

This study follows the guidelines for conducting a literature review using three digital databases: IEEE Xplor, Science Direct, and Scopus. Science Direct makes a highly reputable magazine in science and technology available to the public. Updated research papers in computer science, electronic engineering, engineering, and computer technology applications in medical applications can be found in IEEE Explore. Scopus is a trusted resource in various fields, including medicine, health, science, technology, and engineering. The findings of this literature review can aid in the security of applications by identifying all forms of attacks, including SQL injection attacks, and determining how to prevent attacker access to databases and the theft of user information.

Search strategy

a comprehensive literature search was done in the three databases listed published from 2011 to 2021. These indices were chosen because they provided adequate coverage of studies relating to this research, considering the numerous SQL injection methods and development. Under the notion of AI and ML, DL, this study presented and conducted a Boolean search strategy utilizing various keywords linked to pervasive 'SQL injection' and keywords relevant to the detection, diagnosis, and classification of SQL. These query approaches are used to improve the search of various AI and machine learning systems and application studies for SQL.

Study selection

This method began with removing duplicate articles and screening non-duplicated articles based on their titles and abstracts to ensure that they met the research's inclusion and exclusion criteria. The relevant publications were read thoroughly to collect and extract research data and create the review article from all of the research articles listed.

Data extraction and classification

Looking at the topic, it is concerned with the security aspect of web pages and networks. Various methods have been extracted to protect websites from SQLI attacks using ML, artificial intelligence, and DL to assess the risks of injection attacks and speed up the detection process of these attacks. Information and data were extracted from articles and research over the previous years. These tuners include the author's name and date of publication. This study includes a comprehensive view of SQL injection attacks and ways to detect this type of attack based on machine learning and artificial intelligence.

Inclusion criteria

1. artificial intelligence and machine learning applications, systems, algorithms, methodologies, and techniques emphasize the object.
2. The development focuses solely on SQL injection detection, diagnosis, and categorization.

Table 1: summaries the Boolean search query sequences and results utilized in this paper

Seq.	Query Details	Terms Result of Databases	Final Results

Query1	("Artificial intelligence" OR "Machine learning" OR "Deep learning ") AND ("SQL injection")	SD = 203
		IEEE = 73
			276-12(duplicate)=264

Query2	("SQL injection") AND ("Machine learning")	SD = 54
		IEEE = 32
			86-10(duplicate)=76

5. Result

Figure 2 depicts the results of the search queries used in this investigation. Two search queries were completed to cover all databases and their search engine processes during data gathering. The first result consisted of 362 articles from two databases, including 22 duplicates. After screening the papers based on title and abstract, the final step was to study all critical articles in-depth, with just 20 articles meeting the goal's inclusion criteria.

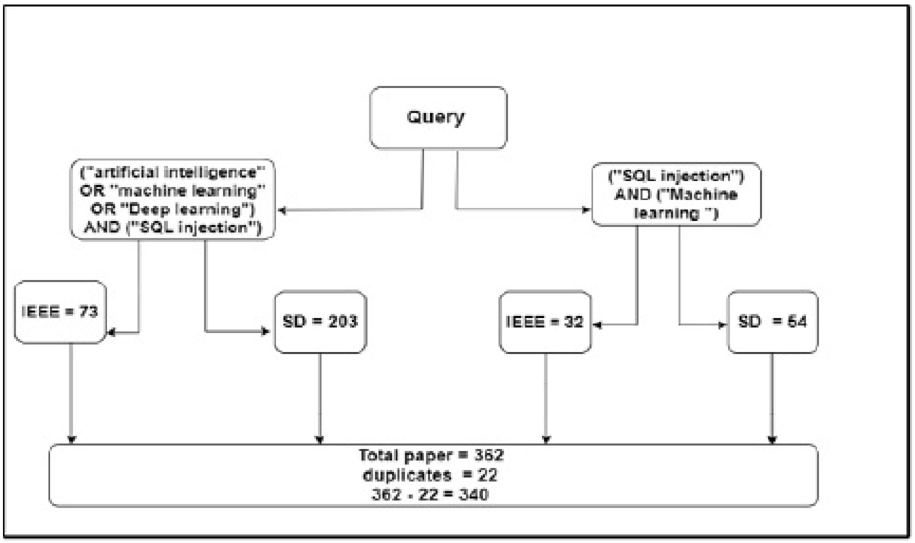
Query
("artificial intelligence" OR "machine learning" OR "Deep learning") AND ("SQL injection")
("SQL injection") AND ("Machine learning ")
IEEE = 73
SD = 203
IEEE = 32
SD = 54
Total paper = 362
duplicates = 22
362 - 22 = 340

Figure 2: Diagram of the result the used queries

Table 2: The table below shows all the sources used in this research paper. The following is the descriptive part of all the sources.

Ref.	Author	Method	Accuracy	Algorithm
[4]	**M.Hasan, Z. Balbahaith, M. Tarique**	Collected SQL injection or non-injection statement in a spreadsheet, pre-processing, analyses the statements, five-fold for the cross-validation, the dataset is used for both training and testing, classification algorithms, available with MATLAB,	93.8%. from Boosted Trees and Bagged Trees	KNN, Bagged Trees, Linear SVM, RUS Boosted Trees, Subspace Discriminant, Boosted Trees, Weighted KNN, Cubic KNN, Linear Discriminant, Medium Tree, Subspace KNN, Simple Tree, Quadratic Discriminant, Subspace KNN, Simple Tree, Quadratic Discriminant, Logistic Regression, Coarse Gaussian, SVM
[5]	**K. Zhang**	Training Dataset, Feature Extraction, Classifier training,Ten-fold cross-validations, IVS and bag-of-words, training classifier by using word2vec feature set	95.4% from CNN	Decision tree, Random forest, SVM, Logistic Regression, Multilayer Perceptron RNN, LSTM, CNN
[6]	**S.S. Anandha Krishnan**	Datasets pre-processing,NLP techniques(parser, feature Extraction), Machine learning algorithm, classified results	97% From CNN	Naive Bayes, Logistic Regression, Passive Aggressive, SVM, CNN
[7]	**K. Kamtuo and Ch. Soomlek**	Commands for SQLI Extraction of datasets, pre-processing, analysis of ML models for SQLI prediction and detection, training and testing	0.9968 and time =2.4725 s from Decision jungle	SVM, Boosted Decision tree, decision tree, Artificial neural network
[8]	**K. Ross, M. Moh, J. Yao, and T.S.Moh**	Traffic creation, data capture, and data pre-processing	RF Web app=96.525% Datiphy =97.210% Correlated=98.055%	Grip, j48 ,RF ,SVM ,ANN
[9]	**D. Tripathy, R. Gohil, and T.Halabi**	Define Problem, Data collection, Data Cleaning, Feature Engineering, Select Algorithm, Training set, Pre-Processing data, training, Model Evaluation, and Predication	99.8% from Random Forest	Tensor Flow Boosted Tree Classifier, Ada Boost Classifier,Random Forest, Decision Tree, SGD Classifier, Deep ANN, Tensor Flow Linear classifier
[10]	**Q. Li, W. Li, J.Wang, and M.Cheng**	1-offline training stage: first collected the training data, then decoded the encoded samples using the pre-processing data module, and then input the feature vectors into the feature extraction module, which is utilized as the deep forest model's input. 2- In the online testing stage, an unknown type of SQL statement should be decoded first by the pre-processing data module, then entered into the feature extraction module to construct the vectors with the exact dimensions as in the offline training stage. Finally, the trained model classifies the SQL statement as harmful or normal.	Higher accuracy with runtime is 33.8 s	Random forest, CNN

[11]	**A. Alam, M. Tahreen, M. M. Alam, S. A. Mohammad, and S. Rana**	Data preprocessing, Training data, Testing data, Algorithm Evaluation	97.8% from Naïve Bayes	Random Forest, KNN, Naïve Bayes, Logistic Regression, Neural Network Classifier
[12]	**P. Tang, W. Qiu, Z. Huang, H. Lian, G. Liu**	Model Evaluation,Model Training(MLP, LSTM), Data Processing	MLP=99.67% LSTM99.17%	CNN
[13]	**A. Luo, W. Huang, W. Fan**	Payloads for SQLI attacks are extracted from the network. traffic and propose a Convolutional Neural Network-based SQL injection detection model (CNN)	0.9950	CNN
[14]	**J. Patel , N. Gandhi, Sisodiya**	SQLI attack detection using a hybrid CNN-BiLSTM technique.	98%	CNN-BiLSTM
[15]	**Xie, X, Ren, C, Fu, Y Xu, J, Guo, J**	The article compares and contrasts a method of SQLI detection based on Elastic-Pooling CNN (EP CNN) with existing detection methods.	CNN Training Set =0.99950 Testing Set = 0.99413 EP-CNN Training Set = 0.9998 Testing Set = 0.99936	CNN , EPCNN, SVM, Naive Bayes, Random Forest, Decision Tree

6. Discussion

Several works have been done to prevent the attacks on SQL injections; below is a review of all previous studies related to SQL injection attacks and the strengths and weaknesses of each method. They define SQL injection as a means for hackers to use a web-based application to execute malicious SQL queries on a database server. In the publication, they also discuss the approach for combating SQL injection and the method for combating SQL injection [4]. M. Hasan, Z. Balbahaith, M.Tarique proposed a machine learning-based heuristic algorithm to prevent SQL injection attacks. The study focuses on collecting a set of features from SQL statements and analyzing them to detect if harmful commands have been inserted into them or not. 23 different machine learning classifiers were trained and tested using a dataset of SQL statements. Based on detection accuracy, choose the top five classifiers from among these and create a Graphical User Interface (GUI) application using these five classifiers. The suggested system has been thoroughly evaluated. The findings demonstrate that employing both Ensemble Boosted and Bagged Trees classifiers, the proposed algorithm can detect SQL injection attacks with a high degree of accuracy (93.8 percent) [1]. K. Zhang presented. The primary goal is to develop a machine learning (ML) based classifier to identify SQLI vulnerabilities files. This is an ML classifier that can detect SQLI flaws in PHP code. Using input validation and sanitization features collected from source code files, both classical and learning-based ML techniques were utilized for training and assessing classifier models. On ten-cross-validation, a model trained with a Convolutional Neural Network (CNN) had the highest precision (95.4%), while a model based on Multilayer Perceptron (MLP) had the highest recall (63.7%) and the highest f-measurement(0.746) [5]. S.S.A. Krishnan, A.N. Sabu, P. P. Sajan, A.L. Sreedeep Presented the SQL Injection Detection Machine Learning Algorithms using a classification method. The classification algorithm identifies the next traffic as SQL Injection or plain text. The problem is classified using five machine learning techniques: Naive

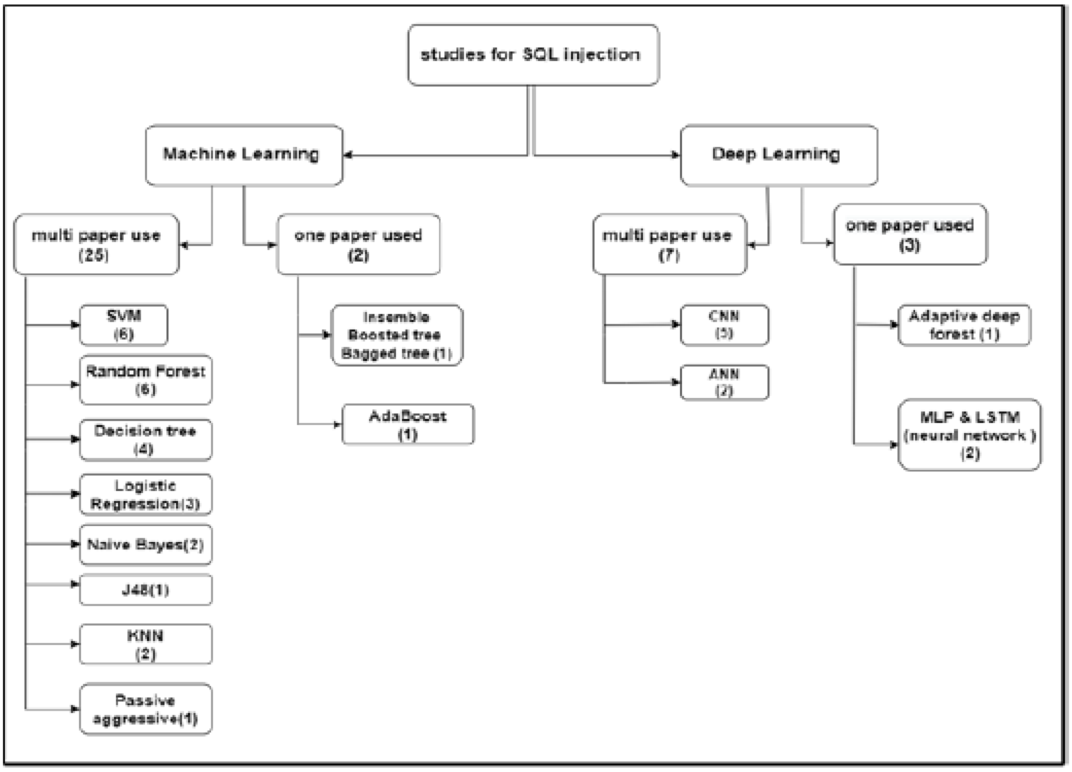

Figure 3: Taxonomy of Studies for SQL injection Used in this paper

Bayes Classifier, Passive Aggressive Classifier, SVM, CNN, and Logistic Regression. The Passive-Aggressive Classifier has a 79 percent accuracy rate, SVM has a 79 percent accuracy rate, and Logistic Regression has a 92 percent accuracy rate. In comparison, the Naive Bayes classifier machine learning model has a 95 percent accuracy rate. Supervised learning methods are thought to deliver outcomes with finer accuracy since they use many fundamental classifiers to increase error and accuracy. Consequently, CNN was picked to solve the SQL Injection categorization problem out of all the algorithms considered. By simultaneously tweaking and testing a variety of parameters, the CNN algorithm has a 97 percent accuracy [6]. K. Kamtuo and Ch. Soomlek presented the purpose of presenting a system for SQL injection prevention on server-side scripting using the compiler platform and machine learning in Illegal and Logically Incorrect Queries, Union Queries, and Piggy-backed Queries. Jungle is the best machine learning model with Pd = 0.9955 SD = 0.0078, Pf = 0 SD = 0, Pr = 1.000 SD = 0, Acc = 0.9968 SD = 0.0054, and Processing Time = 2.4725 Seconds SD = 0.2044 in terms of processing time. Future experiments can aid in the creation of a compiler platform based on an Integrated Development Environment (IDE) that can check SQL syntax and assist detection of SQL injection using Machine Learning in server-side scripts during the development process. The results showed that the decision tree was effective and the best model in terms of processing time, highest efficiency in prediction [7]. K. Ross, M. Moh, J. Yao, and T. Moh proposed a technique for analyzing data from several sources for improved accuracy in detecting SQLI attacks, and found that the algorithms tested, such as rule-based and decision tree algorithms, achieved accuracy close to that of Neural Networks and are much faster when it comes to building models and executing them when classifying testing data collected traffic from two sources: at the web application host, and a Datiphy appliance node located between the web app host and the associated MySQL database server. [8]. D. Tripathy, R. Gohil, and T. Halabi presented for cloud-based software applications and services platforms, Software as a Service (SaaS) has been rapidly accepted. The popularity of SaaS in cloud computing can't mask the security issues that cloud SaaS web applications face. This study aims to look at the possibility of utilizing ML techniques to identify SQL injection at the application level. The algorithms that will be put to the test are classifiers that have been trained on a variety of harmful and Payloads that are not harmful. They examine a payload to see whether it contains any potentially hazardous code. According to the findings, with a detection rate of above 98 percent, these systems can identify conventional payloads from malicious payloads. In addition, the article evaluates the effectiveness of various ML models in identifying SQLI threats. When it comes to detecting malicious code, there are a few factors to consider: the number of different bytes, the length of the input, and the number of punctuation letters. This research looked into which attributes are excellent and poor to employ in machine learning classifiers [9]. Q. LI, W. LI, J. WANG, AND M. CHENG presented an adaptive deep forest-based approach for detecting complicated SQL injection. First, the deep forest's structure is optimized; The input of each layer of the raw feature vector is concatenated with the average of previous outputs. There are two stages to the proposed deep forest-based SQLI detection system: offline training and online testing. The offline training stage begins with using honeypot technology and SQLI samples to acquire training data on the vulnerability submission site. Before transferring the encoded samples to the feature extraction module, the pre-processing data module decodes them. The length of the SQL injection statement, common SQL injection phrases, the feature extraction strategy are used to feed feature vectors into the deep forest model. The data preparation module should first decode the unknown type SQL statement before feeding it to the feature extraction module to create the vector of the same dimension as the offline training stage. Finally, the trained model classifies the SQL statement as harmful or normal. Experiments show that the proposed strategy efficiently handles the problem of deep forest original features degrading as the number of layers increases. Then, a deep forest model based on the AdaBoost algorithm was introduced. It updates the weights of features on each layer based on the error rate. That is, distinct attributes are allocated varying weights in the training process based on their impact on the outcome. This model may automatically change the structure of the tree model. The experiments reveal that the suggested strategy outperforms both traditional machine learning and deep learning methods [10]. A. Alam, M. Tahreen, Md. Alam, Sh. Mohammad, Sh. Rana opted to train models using machine learning algorithms to detect SQL injection known attempts and even discover novel attack methods. They were attempting to train this model using a dataset that contained a sufficient number of SQL injection attack samples and data from regular users. The test datasets were then fed into the model to see how accurate it was. For this challenge, they chose the technique that had the highest accuracy in detecting SQL injection threats. This approach acts as a deterrence to any query that could be harmful to the database. It protects against SQL injection attacks and alerts the server's appropriate authorities. This approach can be used to complement any type of website's security procedures. Finally, the most successful model was combined with the creation of a website. Every input field is connected to the model, allowing it to give total protection against SQL injection attempts across the entire website. This model can be used to safeguard the databases of a variety of web app. It is expected to provide an additional layer of security to database servers in any system. It should be a low-cost but effective alternative to numerous expensive and complex SQLI defenses [11]. This paper provides a neural network-based approach for detecting SQL injection with excellent accuracy. SQL injection, a prevalent web attack, has proven a difficult network security issue, prompting statistical analysis of normal and SQL injection data. There are eight different sorts of characteristics, as well as an MLP model has been constructed based on the statistical data. The model's accuracy remains at around 99 percent. Meanwhile, when the training impact of other ML algorithms (such as LSTM) is compared and analyzed, the results suggest that this technique outperforms the relevant ML algorithms in terms of accuracy. This research provides a neural network-based SQL injection detection framework that detects SQL injection using various neural networks. The MLP-based SQL injection detection model as the neural network's input extracts the relevant URL attributes, then performs MLP network training, saves the model with the best training effect as the final model. The LSTM-based SQL injection detection model turns the URL into a vector and utilizes the vector as the model's training input. The feature extraction method suggested in this paper outperforms previous neural network models in terms of experimental outcomes [12]. A. Luo, W. Huang*, W. Fan presented research papers about detection SQL injection by CNN. Traditional methods of SQL injection detection methods that are inefficient in dealing with the SQL injection techniques are constantly evolving, with the possibility of bypassing versions always present. Extracted SQL injection attack payloads from network traffic and presented a Convolutional Neural Network-based SQL injection detection model (CNN) that can cope with this issue by using the high-dimensional aspects of SQL injection behavior. The suggested method was compared against Mod-Security, a representative rule-matching-based method, in a real-world case study. The outcomes of the experiment reveal that the CNN-based model has a greater accuracy, precision, and recall rate, indicating that it is more successful at detecting attacks and is more resistant to obfuscation [13]. N. Gandhi, J. Patel, and R. Sisodiya, N. Doshi, S. Mishra presented research papers about SQLI attack detection using a hybrid CNN-BiLSTM technique. When compared to previous machine learning algorithms, the suggested CNN-BiLSTM model exhibited a considerable accuracy of 98 percent and higher performance. The key problem with SQLI attacks was the hackers creating new risky SQL queries in order to launch SQLI attacks. Machine learning techniques can be used to efficiently forecast SQLI attacks, which solves the problem.A comparison of different types of ML techniques used for SQL injection attack detection is presented in this research. To summarize, the CNN-BiLSTM based hybrid technique for SQL injection is more accurate than any other ML algorithm reported. By predicting SQL injection attacks, the proposed hybrid CNN-BiLSTM based machine learning model contributes to machine learning by reducing the frequency of SQL injection assaults [14]. X. Xie, C. Ren, Y. Fu, J. Xu, and J. Guo presented an Elastic-Pooling CNN (EP CNN) that is used to identify SQL injections, and it is compared to traditional detection methods. The EP-CNN-based detection of SQL injection extracts the common hidden features of SQL injection. It recognizes the attack flow, bypassing the conventional SQL injection to achieve the highest accuracy in CNN Training Set =0.99950, Testing Set = 0.99413 EP-CNN, Training Set = 0.99980, Testing Set = 0.9993. It has a limited vocabulary and is based on a single character, but it can keep all query statement credentials. If the vocabulary is limited, the training complexity and cost can be decreased. This method efficiently detects SQL injection in web applications by producing a two-dimensional matrix with no data truncation. It can detect new attacks and is more difficult to defeat because of the irregular matching qualities [15].

Motivations

Because of the massive impact of social, scientific-based applications on human life in terms of research fields in the field of AI, such as ML and deep learning technique-based applications, they have been rapidly developed up to this point. As a result, this section explains why studies on SQL injection were done. Machine learning for SQL injection diagnosis systems is effective and can stop the spread of SQL injection attacks and protect people from them. Another form of attack can also be estimated and predicted using machine learning. The Query can be identified and predicted using machine learning techniques. As a result, research into determining the optimum model can aid in reducing the impact of SQL injection attacks in online applications and detecting all sorts of SQL injections that are damaging.

7. Conclusion

The main objective of this paper is to review previous studies on SQL injection attacks and the risks of these attacks on web pages and applications. The other objective is to know the latest studies on the solutions of SQL injection attacks and ways to address them to avoid exposure to this type of attack and provide a safe environment for use on the Internet. The database's SQL queries are discussed to distinguish between malicious and normal. SQL injection is the most popular method hackers use to get sensitive data and information from users. Excessive privilege abuse, justified privilege abuse, privilege elevation, and platform vulnerabilities are all examples of database dangers, as well as methods to deal with them. Previous studies have identified ways to detect SQL injection attacks that give more accuracy and less time to detect maliciously; SQL queries are also covered.

Acknowledgment

We would like to express our gratitude to researcher Mustafa Mohammed Altaee, who assisted us in making this study a success.

References

[1] M. Hasan, Z. Balbahaith and M. Tarique, *Detection of SQL injection attacks: A machine learning approach*, Int. Conf. Electr. Comput. Technol. Appl. (ICECTA), 2019, p. 1-6.

[2] P. Ongsulee, *Artificial intelligence, machine learning, and deep learning*, 15th Int. Conf. ICT Knowledge Engin. 2017, p. 1–6.

[3] M.T. Muslihi and D. Alghazzawi, *Detecting SQL injection on web application using deep learning techniques: A systematic literature review*, Third Int. Conf. Vocational Edu. Electr. Eng. (ICVEE), 2020, p. 1–6.

[4] N. Singh, M. Dayal, S.R.Raw and S. Kumar, *SQL injection: Types, methodology, attack queries and prevention*, 3rd Int. Conf. Comput. Sustainable Global Dev. (INDIACom), 2016, p. 2872–2876.

[5] K. Zhang, *A machine learning based approach to identify SQL injection vulnerabilities*, 2019 34th IEEE/ACM Int. Conf. Autom. Software Eng. (ASE), 2019, p. 1286-1288.

[6] S.S.A. Krishnan, A.N. Sabu, P.P. Sajan and A.L. Sreedeep *SQL Injection detection using machine learning*, Rev. Gest ao Inovação e Tecnol. 11(3) (2021) 300–310.

[7] K. Kamtuo and C. Soomlek, *Machine learning for SQL injection prevention on server-side scripting*, 2016 Int. Comput. Sci. Engin. Conf. IEEE, 2016, p. 1–6.

[8] K. Ross, M. Moh, J. Yao and S.T. Moh, *Multi-source data analysis and evaluation of machine learning techniques for SQL injection detection*, Proc. ACMSE 2018 Conf., vol. 2018-Janua, 2018, p. 1-8.

[9] D. Tripathy, R.Gohil and T. Halabi, *Detecting SQL injection attacks in cloud SaaS using machine learning*, 2020 IEEE 6th Intl Conf. Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Int. Conf. Intell. Data Secur. (IDS), 2020, p. 145-150.

[10] Q. Li, W. Li, J. Wang and M. Cheng, *A SQL injection detection method based on adaptive deep forest*, IEEE Access 7 (2019) 145385–145394.

[11] A. Alam, M. Tahreen, M. Alam,S. A. Mohammad and S. Rana, *SCAMM: Detection and prevention of SQL injection attacks using a machine learning approach*, Doctoral dissertation, Brac University, 2021.

[12] P. Tang, W. Qiu, Z. Huang, H. Lian and G. Liu *Detection of SQL injection based on artificial neural network*, Knowledge-Based Syst. **190** (2020), 105528.

[13] A. Luo, W. Huang and W. Fan, *A CNN-based approach to the detection of SQL injection attacks*, 2019 IEEE/ACIS 18th Int. Conf. Comput. Inf. Sci. (ICIS), 2019, p. 320-324.

[14] N. Gandhi, J. Patel, R. Sisodiya, N. Doshi and S. Mishra *A CNN-BiLSTM based approach for detection of SQL injection attacks*, Proc. 2nd IEEE Int. Conf. Comput. Intell. Knowl. Econ. ICCIKE 2021, p. 378-383.

[15] X. Xie, C. Ren, Y. Fu, J. Xu and J. Guo *SQL injection detection for web applications based on elastic-pooling CNN*, IEEE Access **7** (2019), 151475–151481.

[16] *SQL injection attacks-web-based app security, part 4 spanning*, `https://spanning.com/blog/sql-injection-attacks-web-based-application-security-part-4/` (accessed Dec. 30, 2021).

[17] *OWASP top ten web application security risks OWASP*, `https://owasp.org/www-project-top-ten/` (accessed Jan. 04, 2022).

[18] *What is SQL injection SQLI attack example & prevention methods imperva*, `https://www.imperva.com/learn/application-security/sql-injection-sqli/` (accessed Dec. 30, 2021).

[19] *World internet users statistics and 2021 world population stats*, `https://www.internetworldstats.com/stats.htm` (accessed Jan. 03, 2022).