www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 02, February 2025, pp : 281-289

e-ISSN : 2583-1062
Impact Factor : 7.001

INNOVATIVE FRAMEWORK FOR SQL INJECTION DEFENCE: LEVERAGING MACHINELEARNING AND HYBRID APPROACHES FOR ENHANCED DETECTION AND PREVENTION

Shashi Mani Dwivedi[1], Aryan Prajapati[2], Rinku Raheja[3]

[1,2]Student, Computer Science, National PG College, Lucknow, Uttar Pradesh, India.

[3]Assistant Professor, National PG College, Lucknow, Uttar Pradesh, India.

DOI: https://www.doi.org/10.58257/IJPREMS38508

ABSTRACT

A web operation is a software system that provides an interface to its druggies through a web cybersurfer on any operating system. Despite their growing fashionability, web operation security pitfalls have come more different, performing in more severe damage. Malware attacks, particularly SQLI attacks, are common in inadequately designed web operations. This vulnerability has been known for further than two decades and is still a source of concern. Consequently, different ways have been proposed to fight SQLI attacks. still, the maturity of them either fail to cover the entire compass of the problem. The structured query language injection (SQLI) attack is among the most dangerous online operation attacks and frequently happens when the bushwhacker alter (modify), remove (cancel), read, and dupe data from database waiters. All angles of security, including confidentiality, data integrity, and data vacuity, can be impacted by a successful SQLI attack. This paper investigates common SQLI attack forms, mechanisms, and a system of relating, detecting, and precluding them grounded on the actuality of the SQL query. Then, we've developed a comprehensive frame for detecting and precluding the effectiveness of ways that address specific issues following the substance of the SQLI attacks by using traditional Navies Bayes (NB), Decision Trees (DT), Support Vectors Machine (SVM), Random timbers (RF), Logistic Retrogression (LR), and Neural Networks Grounded on Multilayer Perceptron(MLP), and cold-blooded approach are used for our study. The machine literacy(ML) algorithms were enforced using the Keras library, while the classical styles were enforced using the Tensor Flow Learn package. For this proposed exploration work, we gathered 54,306 pieces of data from weblogs, eyefuls, session operation, and from HTTP(S) request flees to train and test our model. The performance evaluation results for training set in criteria similar as the mongrel approach(ANN and SVM) perform better rigor in perfection(99.05 and 99.54), recall(99.65 and 99.61), f1-score(99.35 and 99.57), and training set(99.20 and 99.60) independently than other ML approaches. still, theirtraining time is too high(i.e., 19.62 and 26.16 s independently) for NB and RF. Consequently, the NB fashion performs inadequately in delicacy, perfection, recall, f1- score, training set evaluation criteria , and stylish in training time. also, the performance evaluation results for test set in criteria similar as cold-blooded approach(ANN and SVM) perform better rigor in perfection(98.87 and 99.20), recall(99.13 and 99.47), f1- score(99.00 and 99.33) and test set(98.70 and 99.40) independently than other ML approaches. still, their test time is too high(i.e., 11.76 and 15.33 ms independently). Consequently, the NB fashion performs inadequately in delicacy, perfection, recall, f1- score, test set evaluation criteria , and stylish in training time. Then, among the enforced ML ways, SVM and ANN are weak learners. The achieved performance evaluation results indicated that the proposed SQLI attack discovery and forestallment medium has been bettered over the preliminarily enforced ways in the theme. Eventually, in this paper, we aimed to keep experimenters up - to- date, with benefactions, and recommendations to the understanding of the crossroad between SQLI attacks and forestallment in the artificial intelligence(AI) field.

Keywords Deep literacy, Discovery, mongrel, Machine literacy, Prevention, SQLI attack, Web operation

1. INTRODUCTION

Malware attacks, particularly SQLI attacks, are common in inadequately designed web operations. This vulnerability has been known for further than two decades and is still a source of concern (1). For numerous times, structured query language(SQL) has been the assiduity standard for dealing with relational database operation systems(DBMS). Since the maturity of operations for cyber-physical systems are safety – critical; misbehavior brought on by arbitrary crimes or online attacks can oppressively limit their development (2, 3). thus, it's pivotal to guard cyber-physical systems from suffering this kind of attack. SQLI attacks on data- driven web operations and systems, also known as SQLI attacks, have been a serious problem since it came common for internet web operations and SQL databases to be connected(4, 5, 6). An SQLI attack occurs when an bushwhacker takes advantage of a excrescence in the web operation's SQL perpetration by submitting a vicious SQL statement through a fillable field. In other words, the bushwhacker will fit law into a field to leave or alter data or gain access to the backend. As explained by(6) and(7), SQLI is a common attack vector that allows vicious SQL law to pierce retired information by manipulating database back ends and is regarded as

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 02, February 2025, pp : 281-289
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

one of the most dangerous injection attacks because it jeopardizes the main security services similar as confidentiality, authentication, authorization, and integrity(8, 9). This information could include sensitive business information, private client information, or stoner lists. A successful SQLI bushwhacker can lead to the omission of entire databases, the unauthorized use of sensitive data, and the unintentional entitlement of executive rights to a database. The increased development and spread of web operations have also increased the number and inflexibility ofweb attacks(10, 11). According to(12), the most common vulnerability in web operations is injection. Injection attacks take advantage of a variety of excrescencies to deliver untrusted stoner input, which is also reused by a web operation(13). The SQLI attacks number edging in(fitting) vicious SQL commands into input forms or queries to gain access to a database or manipulate its data(e.g. shoot the database contents to the bushwhacker, modify or cancel the database content, etc.)(14, 15). incontrovertibly, utmost web operations moment calculate on a back- end database to store datacollected from druggies and/ or to recoup information named by druggies(16). The SQL query that was virulently fitted is intended to prize or modify data from the database garçon. Successful injection can beget data loss and/ or the total database to be destroyed, as well as authentication, bypass, and variations to the database by fitting , changing, and/ ordeleting data. also, such an assault could take control and run commands on it, generally having lesser negative goods(2, 18). thus, associations are seriously hovered by SQLI assaults. Indeed though several ways have been proposed to combat SQLI attacks, none of these results have addressed the full compass of the attacks. As a result, there were no results that can help or descry all types of SQLI attacks. lately, experimenters have tried to with AI integrated ways including deep literacy(DL), machine literacy(ML), and cold-blooded ways to propose more sophisticated results(19). Then, learning from once data reflecting an attack and/ or regular data is generally used to make AI approaches to help with trouble discovery and forestallment. literal information can be used to interpret detected business, identify attack patterns, and indeed read unborn assaults before they be(2, 20).

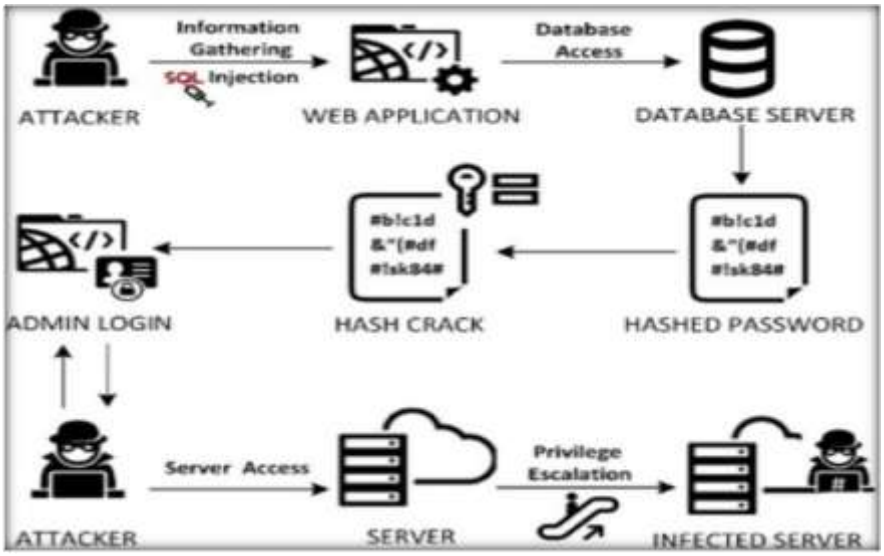

Fig:1 The SQL Attacks

Consequently, this work has the following major benefactions.

- Review the recommended and state- of- the- art exploration works in DL, ML, and cold-blooded ways for SQLI attacks that have been published in estimable databases.
- The different SQLI attack discovery and forestallment countermeasures are classified and bandied Studying and relating the nature of SQLI attacks and proposing forestallment and discovery mechanisms.
- recently proposed results are described and bandied, similar as those grounded on DL, ML, and cold-blooded ways.
- Keep the experimenters up- to- date, with benefactions, and recommendations to the understanding of the crossroad betweenSQLI attacks and forestallment in the AI field. Consequently, the primary thing of this paper is to examine current SQLI attacks, identify their methodologies, strengths, and sins, and eventually propose a thorough discovery and forestallment system. The rest of the paper is organized into different but inter related sub-sections. The paper begins by agitating the affiliated workshop in the " Affiliated workshop " section, the SQLI query attacks overview in the" SQLI query attacks overview" section, being styles for SQLI discovery and forestallment in the" Being styles for SQLI discovery and forestallment" section, developing a web- grounded frame for SQLI attacks discovery and forestallment in" Developing a web- grounded frame for SQLI attacks discovery and forestallment"

section, most common attacks on SQLI in" Most common attacks on SQLI" section, proposed fabrics for SQLI discovery and forestallment in" Proposed fabrics for SQLI discovery and forestallment" section, result and discussion in" Result and discussion" section, and the conclusion and recommendation in" Conclusion and recommendation" section. Affiliated workshop In this section, different published exploration workshop have been considered and included to indicate the exploration gaps in the area. The paper generally reviews and includes studies that have been published in estimable databases. numerous experimenters have demonstrated the use of DL, ML, and cold-blooded ways to descry SQLI attacks(23). A review of SQLI forestallment in web operations has been presented in(1). The authors have handed a summary of 14 different kinds of SQLI attacks and how they affect online operations. Their exploration's main ideal was to probe indispensable SQLI forestallment strategies and to offer an analysis of the most effective defense against SQLI attacks. Authors in(2) have conducted a methodical literature review of 36 papers related to exploration on SQLI attacks and ML ways. To classify different kinds of SQLI attacks, they've linked the most extensively used ML ways. Their finding revealed that many studies generated new SQLI attack datasets using ML tools and ways. also, their results showed that only a many studies concentrated only on using mutation drivers to induce inimical SQLI attack queries. In unborn work, the experimenters aimed to cover the use of other ML and DL ways to induce and descry SQLI attacks. A comprehensive study on SQLI attacks, their mode, discovery, and forestallment has been presented in(4). The authors have linked how bushwhackers of this kind might exploit such a weakness and execute weak law as well as a strategy to alleviate similar mischievous goods on database systems. The experimenters' disquisition revealed that web operations were constantly used for online administrations ranging from high situations of informal communication to managing sale accounts and dealing with sensitive stoner data. The real issue, still, was that this data was exposed to attacks because of unauthorized access, where the bushwhackers gained entry to the system using colorful hacking and cracking ways with veritably vicious motives. The bushwhacker can use more sophisticated queries and creative tactics to get around authentication while also gaining total control over both the garçon and the web operation. numerous slice- edge algorithms have been developed up to this point to encrypt data queries to defend against similar attacks by structuring desirable query revision plans. In the paper, they worked together to bandy the history of injection attacks, different forms of injection attacks,colorful case studies, and defenses against SQLI attacks, along with an applicable illustration

SQLI query attacks overview

An SQL is a language developed to manage data stored in relational databases(12). It allows druggies to pierce, modify, and cancel data. numerous web operations and websites keep all of their data in SQL databases. SQL commands can also be used to run commands in some cases. Web attacks are one of the major motifs to be delved in this study. Indeed though there are multitudinous web attacks, SQLI is one of the most common and will be among the top five web attacks in 2021, according to the OWASP report(55, 56). This attack subventions bushwhackers complete access to databases containing sensitive information(1, 10, 57, 58). As in a common understanding, a web operation has three situations(54) The first Demilie and Deriba Journal of Big Data(2022) 9124 runner 14 of 30 donation subcaste collects stoner feedback and shows the stoner the processing results.

The donation subcaste directly communicates with the stoner. The alternate control subcaste, the garçon script, processes data entered by the stoner and sends the results to the database subcaste. The database subcaste sends the reused data t o the control subcaste, which also sends it to the donation subcaste for the stoner to view(59 – 62). As a result, data processing occurs on the control subcaste in the web operation, which can be enforced in a variety of garçon scripting languages. The database(DB) subcaste eventually saves and retrieves the data. The database stores and manages all sensitive web operation data. Because this subcaste is directly connected to the control subcaste and has no security checks, if the control subcaste is successfully attacked, data in the database can be exposed and modified. The general conception of web- grounded armature is depicted in Fig. 2.

The difficulty in perceiving the fitted query at the database subcaste necessitates a system that controls and filters the query at the donation subcaste grounded on destined parameters(63). colorful studies have been conducted to identify and help the fitted queries. The bulk of them, still, don't identify all types of SQLI, but they fared better on a sprinkle in the statistical or dynamic portions. Vulnerabilities in web operations can live if the sanitization function doesn't rightly sanitize stoner input.

The static analysis can not tell whether or not the input has been sanitized duly. The vulnerabilities frequently go unnoticed due to similar excrescencies in static assessments. The SQL help checks produced queries for those parameters and provides an alarm when a hypertext transfer protocol(s)(HTTP(S)) request parameter influences the syntax structure of a query(64).

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 02, February 2025, pp : 281-289
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

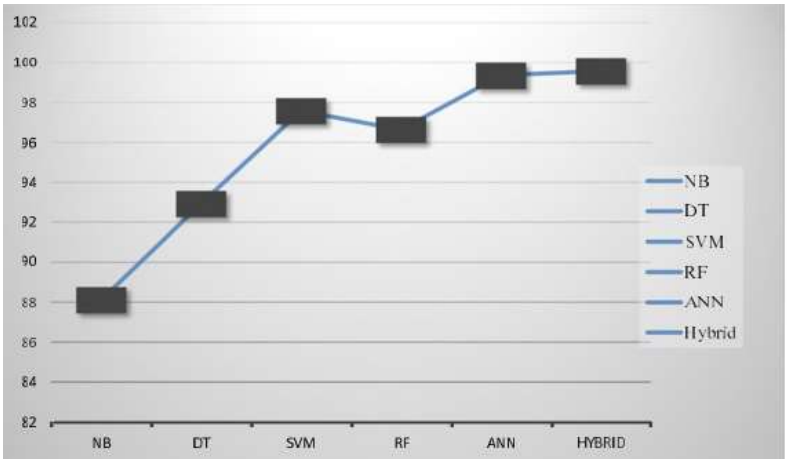

Fig.2 Performance Comparison of Machine Learning Models for Classification Accuracy

Different ways are used to track stoner inputs, and a profile of minding queries has been created. This fashion has a high rate of false cons and false negatives due to the incapability of the operation that generates the query to synopsize input(65). As a result, there's still a gap in the SQLI attack forms. Then, we've proposed this work to fill similar gaps by using the recommended ways and state- of- the- art exploration works consequently.

Existing techniques for SQLI detection and prevention

Several techniques for detecting and preventing SQLI have been proposed, with some focusing on statistical analysis [4, 66, 67, 68, 69, 70] or dynamic analysis [71, 72], others on Hybrid approach [32, 73]. These techniques are used for web applicationvulnerability.

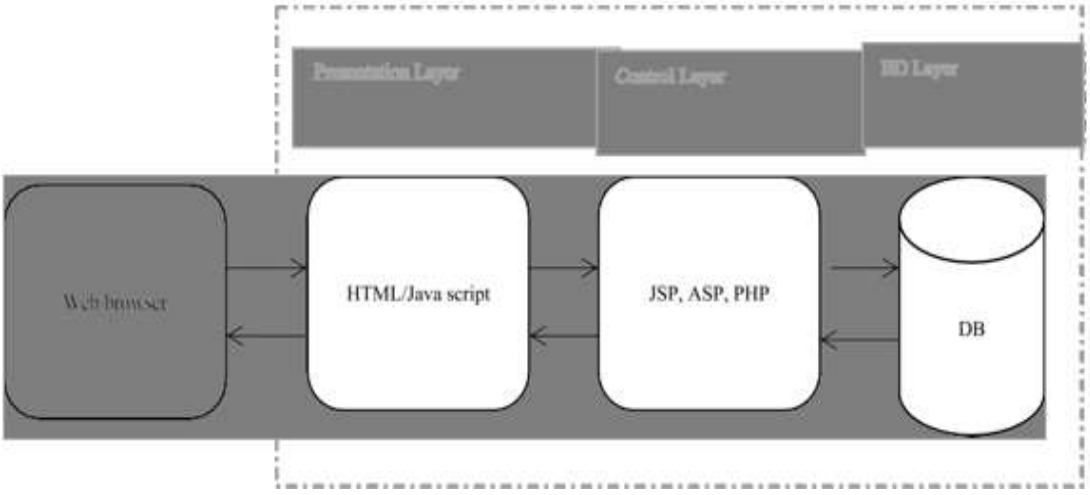

Fig.3 Three-Tier Web Application Architecture: Presentation, Control, and Database Layers

DL ways SQLI is a common and delicate attack on web operations, systems, and network security issues. Deep or convolutional neural networks(CNN), can be used in a wide variety of trouble discovery and forestallment scripts(10, 11). Code injection is the most common and dangerous attack, ranking first on the OWASP vulnerabilities list(16, 17). Consequently, the discovery and forestallment of law injection attacks, which were preliminarily done using hand or pattern- grounded recognition ways, has lately been supplemented by the use of advanced

ML ways. ML ways Several types of exploration inferred that ML ways(12, 77 – 79) can be employed to develop vulnerability predictors. The thing, anyhow of the fashion used, is to learn data associated with injection, which can also be used to prognosticate vulnerability to new injections. A vulnerability analysis system needs to be suitable to acclimatize when more advanced security pitfalls are discovered. The ML fashion allows fore-training to respond to new vulnerability trends(16). mongrel ways The mongrel injection discovery uses both ML classifiers and other statistical ways to help and descry the deliverances of SQLI attacks from different web operations and systems(32). Consequently, some of the former exploration has used cold-blooded ways(64, 73, 80). This can be done by comparing thestructure of the queries to descry attacks. originally, it detected if a stoutly generated query has a different structure or alphabet that meets certain conditions like data length, range, and form by input confirmation and input sanctification by allowing only predefined characters and refusing all others, including those with unique significance to the practitioner than a static query were followed. A new approach is thus demanded for SQLI attacks(18, 81)

www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 02, February 2025, pp : 281-289

e-ISSN : 2583-1062
Impact Factor : 7.001

Most common attacks on SQLI

As we know, SQL is a programming language that's used to produce, update, and access data in a database. A hacker can designedly beget the operation to fail, cancel data, steal data, or gain unauthorized access by precisely casting SQL commands(90). To address the forenamed issue, we give a detailed overview of the colorful types of SQLI attacks discoveredto date. For each type of attack, we give explanations and exemplifications of how similar attacks can be carried out, as well as unequivocal mitigation mechanisms. Eventually, we propose a comprehensive frame that's resistant to all types of attacks for discovery and forestallment Developing a web grounded frame for SQLI attacks discovery and forestallment fabrics have come an essential part of web development because, as web operation norms rise, so does the complexity of the technology needed(82). It's fully unreasonable to resuscitate the wheel with similar sophisticated ways.

As a result, using fabrics championed by thousands of inventors worldwide is a veritably sound approach to developing rich and interactive web operations. Because a web operation has a backend(garçon- side) and a frontend(customer-side) for both the backend and frontend fabrics. numerous fabrics similar as(53, 83, 84) have been developed and tested with colorful parameters.

Authors of(85) proposed a frame grounded on abuse and anomaly discovery ways to descry SQLI attacks. The exploration of(86) discusses a secure medium for guarding web operations from SQLI attacks by using a frame and database fire wall. An author of(81) presents a frame that can be used to handle tautology - grounded SQLI attacks using the post-deployment monitoring fashion. The authors of(87) estimate runtime monitoring fabrics to descry and help SQLI attacks on web operations. The authors of(88) present a pall calculating relinquishment frame(CCAF) security suitable for business shadows. The authors of(89) propose SQLI intrusion discovery frame as a service for SaaS providers, structured query language injection identity as a service, which allows a SaaS provider to descry structured query language injection attacks(SQLIAs) targeting several SaaS operations without reading, analyzing, or modifying the source law. To raise the tenants' mindfulness of the soberness of SQLIAs.

The exploration work of(83) introduces a new business- grounded SQLIA discovery and vulnerability analysis frame named(DIAVA), which can proactively shoot warnings to tenants instantly. Some of them perform well on the given parameters, while others do not. As a result, while these fabrics descry fitted queries, they've no control over them. st ill, a near examination of the literature on the afore mentioned SQLI attack reveals multitudinous gaps and failings. Tautology attack

The attacker attempts to use a conditional query argument to test always true in the tautology attack, such as (1 = 1) or (− −).The attacker injects the condition and transforms it into a tautology that is always valid using the WHERE clause [91, 92]. This type of attack is commonly used to access databases without requiring authentication on web sites [1].

For example: The most common type of tautology attack, the nature of the attack, and the approach used to detect them are described below in Table 1.

Table 1 Common tautology attacks [54]

Type of injection	Nature of attack	Approach for detection
String SQLI	Bypassing authentication, identifying injectable parameters using string datatype, extracting data	Rule-based
Numeric SQLI	Bypassing authentication, identifying injectable parameters using numeric data type extracting data	Rule-based
Comment Attack	Bypassing authentication, identifying injectable parameters using the com ment form, extracting data	Rule-based
Type of injection	Nature of attack	Recommended techniques
Union query attack	Bypassing authentication, extracting data using union operation	Rule-based

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 02, February 2025, pp : 281-289
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

Proposed frameworks for SQLI detection and prevention

Because of the nature of the attack and the need for detection and prevention mechanisms, a more systematic and theoretical analysis of SQLI attacks is required. To develop our framework, we have investigated existing techniques, as well as their attacking methods and flaws accordingly. As a result, we propose a comprehensive framework that addresses all vulnerabilities identified in the previous research works.

To carry out the activity, the attacker must first open his browser and if the application is op en, the intruder either enters his password into the application or requests authorization to access the web service via the internet. The intruder must first get past the firewall checker to proceed. The web server then accepts user input through various mechanisms, such as user input validation, and uses the input to generate queries to an underlying database [64, 93]. This can be accomplished by identifying injection parameters, determining the type and version of a web application's database, and determining the database schema.If the attacker was granted permission based on the request, he will request application server access again.

There were several stages preceding the classification of SQL queries. The first feature extraction is done by comparing the static and dynamic analysis to see if the requested queries are injected with either approach. Based on the query, the classifieraccepts it and matches it with the trained dataset. The extracted feature is then accepted by the ML classifier, which trains themodel to identify the injected query.

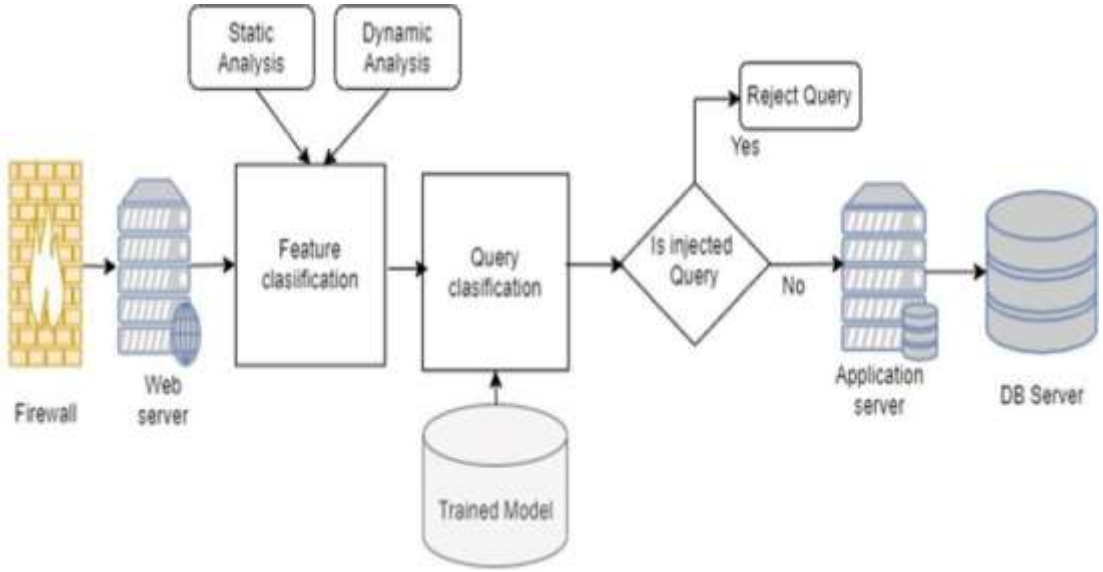

Fig :4 Machine Learning-Based SQL Injection Detection System

The SVM [103, 104], DT, NB [73, 105, 106], and other algorithms in ML techniques [75, 107–111] are used to solve classification algorithms.

The trained model passes all stages such as pre-processing and feature extraction. As a result, during the feature extraction steps, the classifiers will be trained to recognize various types of SQLI attacks based on the given trained ML model and hybrid approach. Based on the trained pre-fetched and trained dataset, the model matches the pattern of each line query requested.

If the SQL query contains one or more qualified attacks, the model will either reject the request or sendit to the application and database servers to perform the requested operation if the query is pure SQL with no injection. As a result, we propose developinga new architecture based on ML and hybrid approaches to achieve the best possible results when dealing with SQLI query attacks.

2. RESULT AND DISCUSSION

In this study, we used three injection parameters in various forms. The first is through a user input field, which allows a web application to use HTTP (S) POST and GET to request information from a backend database, and the second is through cookies, which can be used to restore a client's state information when they return to a web application. An attacker can exploit this vulnerability to change cookies and submit them to the database server if a web application uses the contents of cookies to construct SQL queries. Finally, a server variable can be created by analyzing session usage information and recognizing browsing behaviours. Because attackers can forge the values in HTTP (S) and network headers by entering malicious input into the application's client-end or by crafting their request to the server, logging these variables to a database without sanitization could result in SQLI vulnerability.

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 02, February 2025, pp : 281-289
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

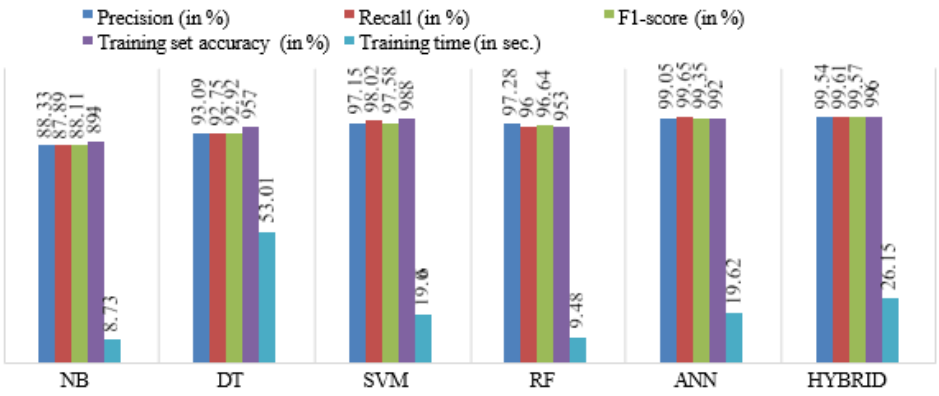

Fig. 5 Performance Evaluation

Accordingly, all the attacks sent to the server are logged and saved as attack log data in the database. Furthermore, attack log data is divided into two categories: attacks and normal data. Using various ML techniques, we trained and assessed vulnerability classifier models to determine which approach performed the best. The set of algorithms includes traditional NB, DT, SVM, RF, LR, and Neural Networks Based on MLP and hybrid techniques that are used for our study. The ML algorithms were implemented using the Keras library, while the classical

methods were implemented using the Tensor Flow-Learn package. We evaluated the performance of the models using ten-fold cross- validations, where the dataset was divided into ten different partitions and the final accuracy result was recorded. During the training and testing of the selected techniques, we can get multiple classifiers, and we need to evaluate the performance of each classifier using appropriate evaluation metrics, from which the best one is selected.

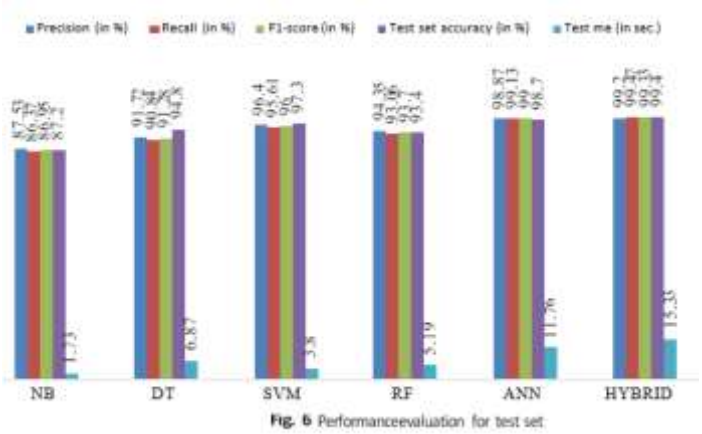

Fig. 6 Performanceevaluation for test set

Proposed Fabrics for SQLI discovery and forestallment

Because of the nature of the attack and the need for discovery and forestallment mechanisms, a more methodical and theoretical analysis of SQLI attacks is needed. To develop our frame, we've delved being ways, as well as their attacking styles and excrescencies consequently. As a result, we propose a comprehensive frame that addresses all vulnerabilities linked in the former exploration workshop. To carry out the exertion, the bushwhacker must first open his cyber surfer andif the operation is open, the meddler either enters his word into the operation or requests authorization to pierce the web service via the internet. The meddler must first get past the firewall checker to do. The web garçon also accepts stoner input through colorful mechanisms, similar as stoner input confirmation, and uses the input to induce queries to an underpinning database(64, 93). This can be fulfilled by relating injection parameters, determining the type and interpretation of a web operation's database, and determining the database schema. However, he'll request operation garçon access again, If the bushwhacker was granted authorization grounded on the request. There were several stages antedating the bracket of SQL queries. The first point birth is done by comparing the static and dynamic analysis to see if the requested queries are fitted with either approach. Grounded on the query, the classifier accepts it and matches it with the trained dataset. The uprooted point is also accepted by the ML classifier, which trains the model to identify the fitted query. The SVM(103, 104), DT, NB(73, 105, 106), and other algorithms in ML ways(75, 107 – 111) are used to break bracket algorithms. The trained modelpasses all stages similar processing and point birth. As a result, during the point birth way, the classifiers will be trained to fete colorful types of SQLI attacks grounded on the given trained ML model and mongrel approach. Grounded onthe trained pre-fetched and trained dataset, the model matches the pattern of each line query requested. However, the model will either reject the request or shoot it to the operation and database

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 02, February 2025, pp : 281-289
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

waiters to perform the requested operation if the query is pure SQL with no injection, If the SQL query contains one or further good attacks. As a result, we propose developing a new armature grounded on ML and cold-blooded approaches to achieve the stylish possible results when dealing with SQLI queryattacks. Result and discussion In this study, we used three injection parameters in colorful forms. The first is through a stonerinput field, which allows a web operation to use HTTP(S) POST and GET to request information from a backend database, and the second is through eyefuls, which can be used to restore a customer's state information when they return to a web

operation. An bushwhacker can exploit this vulnerability to change eyefuls and submit them to the database garçon if a web operation uses the contents of eyefuls to construct SQL queries. Eventually, a garçon variable can be created by analyzing session operation information and feting browsing behaviors. Because bushwhackers can forge the values in HTTP(S) and network heads by entering vicious input into the operation's customer- end or by casting their request to the garçon, logging these variables to a database without sanitization could affect in SQLI vulnerability. Performance evaluation for training set Performance distribution of the ways in the training set(f1- score) Consequently, all the attacks transferred to the garçon are logged and saved as attack log data in the database. likewise, attack log data is divided into two orders attacks and normal data. Using colorful ML ways, we trained and assessed vulnerability classifier models to determine which approach performedthe stylish. The set of algorithms includes traditional NB, DT, SVM, RF, LR, and Neural Networks Grounded on MLP and cold-blooded ways that are used for our study. The ML algorithms were enforced using the Keras library, while the classicalstyles were enforced using the Tensor Flow- Learn package. We estimated the performance of the models using tenfold cross- validations, where the dataset was divided into ten different partitions and the final delicacy result was recorded. During the training and testing of the named ways, we can get multiple classifiers, and we need to estimate the performance of each classifier using applicable evaluation criteria , from which the stylish one is named.

Conclusion and recommendation

SQLI is the most dangerous web application attacker. This type of attacker poses a significant risk to web applications and this may have major implications for privacy and security issues. Web application attacks are becoming increasingly common and severe.

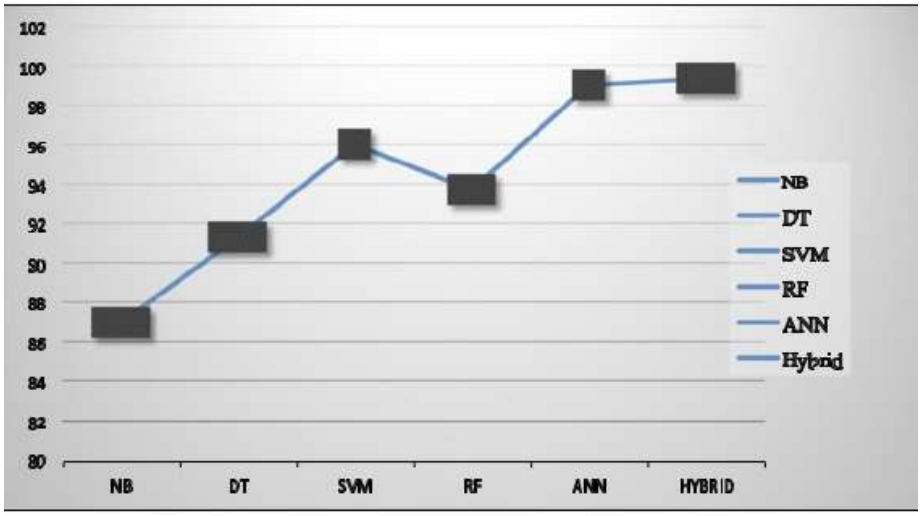

Fig. 7 Performance distribution of the techniques in the test set (f1-score)

A large amount of data available on the internet motivates hackers to launch novel attacks. Several studies have been conducted to mitigate this attack, either by preventing it at an early stage or by detecting it when it occurs. We evaluated various strategies for detecting and preventing SQLI. Firstly, we have defined the different types of SQLI attacks that have been discovered thus far. The techniques under consideration were then evaluated in terms of their ability to detect and prevent SQLI attacks. We identified the most commonly used DL, ML, and hybrid techniques to detect and prevent all types of SQLI attacks. We also looked into the various mechanisms and determined which techniques could deal with the detection and prevention of such SQLI attacks from different web applications. Then, using ML and hybrid techniques, we identify the specifications for each technique and develop a comprehensive framework for detecting and preventing SQLI attacks. We investigated that hybrid and ANN are the best techniques for classifying SQLI based on our model performance evaluation. The performance evaluation results for training set in metrics such as the hybrid approach (ANN and SVM) perform better accuracies in precision (99.05% and 99.54%), recall (99.65% and 99.61%), f1-score (99.35% and 99.57%), and training set (99.20% and 99.60%) respectively than other ML approaches. However, their training time is too high (i.e., 19.62 and 26.16 s respectively) for NB and RF. Accordingly, the NB technique

performs poorly in accuracy, precision, recall, f1-score, training set evaluation metrics, and best in training time. Additionally, the performance evaluation results for test set in metrics such as hybrid approach (ANN and SVM) perform better accuracies in precision (98.87% and 99.20%), recall (99.13% and 99.47%), f1 -score (99.00% and 99.33%) and test set (98.70% and 99.40%) respectively than other ML approaches. However, their test time is too high (i.e., 11.76 and 15.33 ms respectively). Accordingly, the NB technique performs poorly in accuracy, precision, recal l, f1-score, test set evaluation metrics, and best in training time. Here, among the implemented ML techniques SVM and ANNare weak learners. Finally, in this research work, we aimed to keep researchers up -to-date, with contributions, and recommendations to the understanding of the intersection between SQLI attacks and prevention in the AI fi eld. Here, maximizing the dataset and running with different techniques in a real-world environment i s recommended for future researchers.

3. REFERENCES

[1] Johny JHB, Nordin WAFB, Lahapi NMB, Leau YB. SQL Injection prevention in web application: a review. In: Communications in computer and information science, vol. 1487 CCIS, no. January. 2021. p. 568 –585. https://doi.org/ 10.1007/978-981-16-8059-5_35.

[2] Alghawazi M, Alghazzawi D, Alarif S. Detection of sql injection attack using machine learning techniques: a systematic literature review. J Cyber secure Privacy. 2022;2(4):764–77.

[3] Han S, Xie M, Chen HH, Ling Y. Intrusion detection in cyber-physical systems: techniques and challenges. IEEE Syst J. 2014;8(4):1052– 62.

[4] Dasmohapatra S, Priyadarshini SBB. A comprehensive study on SQL injection attacks, their mode, detection and prevention. 2 021. p. 617– 632. https://doi.org/10.1007/978-981-16-3346-1_50.

[5] Hu J, Zhao W, Cui Y. A survey on SQL injection attacks, detection, and prevention. In: ACM international conference on pro ceeding series, no June. 2020. p. 483–488. https://doi.org/10.1145/3383972.3384028.

[6] Blog. What is SQL injection attack? Definition & FAQs|Avi networks.

[7] Imperva. SQL (structured query language) injection. Imperva. 2021.

[8] Deepa G, Thilagam PS, Khan FA, Praseed A, Paris AR, Palsetia N. Black-box detection of XQuery injection and parameter tampering vulnerabilities in web applications. Int J Inf Secure. 2018;17(1):105–20. https://doi.org/10.1007/ s10207-016-0359-4.

[9] Dizdar A. SQL injection attack: real life attacks and code examples. 2022.

[10] Pan Y, et al. Detecting web attacks with end-to-end deep learning. J Internet Serve Appl. 2019. https://doi.org/10. 1186/s13174-019-0115- x.

[11] Zhang W, et al. Deep neural network-based SQL injection detection method. Secure Commune Networks. 2022;2022:1–9. https://doi.org/10.1155/2022/4836289.

[12] Pattewar T, Patil H, Patil H, Patil N, Taneja M, Wadile T. Detection of SQL injection using machine learning: a survey. I nt Res J End Technol (IRJET). 2019;6(11):239–46.

[13] Banach Z. Most dangerous food pathogens. 2022.

[14] Fang Y, Peng J, Liu L, Huang C. WOVSQLI: detection of SQL injection behaviors using word vector and LSTM. In: ACM international conference on proceeding series. 2018. p. 170–174. https://doi.org/10.1145/3199478.3199503.

[15] Li Q, Wang F, Wang J, Li W. LSTM-based SQL injection detection method for intelligent transportation system. IEEE Trans Veh Technol. 2019;68(5):4182–91. https://doi.org/10.1109/TVT.2019.2893675.

[16] Chen D, Yan Q, Wu C, Zhao J. SQL injection attack detection and prevention techniques using deep learning. J Phys Conf Se r. 2021;1757(1):012055. https://doi.org/10.1088/1742-6596/1757/1/012055.

[17] Abaimov S, Bianchi G. A survey on the application of deep learning for code injection detection. Array. 2021;11(June):100077. https://doi.org/10.1016/j.array.2021.100077.

[18] Son S, McKinley KS, Shmatikov V. Diglossia: detecting code injection attacks with precision and ef ficiency. Proc ACM Conf Comput Commune Secure. 2013;2:1181–91. https://doi.org/10.1145/2508859.2516696.

[19] Yan R, Xiao X, Hu G, Peng S, Jiang Y. New deep learning method to detect code injection attacks on hybrid applications. J System Software 2018;137:67–77. https://doi.org/10.1016/j.jss.2017.11.001.

[20] P. Vähäkainu and M. Lehto, " Artificial intelligence in the cyber security environment," Proc. 14th Int. Conf. Cyber Warf. Secur. ICCWS2019 Artif., 2019