International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 06 | June-2016 www.irjet.net p-ISSN: 2395-0072

SQL Injection Attack and User Behavior Detection by Using Query Tree, Fisher Score and SVM Classification

Aniruddh Ladole [1], Mrs. D. A. Phalke[2]

[1]Department of Computer Engineering
D. Y. Patil College of Engineering, Akurdi, Pune, India
Email: aniruddhladole@gmail.com

[2]Department of Computer Engineering
D. Y. Patil College of Engineering Akurdi, Pune, India

Abstract - *Most of the web applications maintain information at the backend database from which results are retrieved. SQL injection is most dangerous threat for the web application and sites. Detecting SQL injection attacks (SQLIAs) is becoming increasingly important in database driven web sites. SQL Injection is the very easy way to attack the application to reveal the data in the data driven application. A lot of research is done to detect and prevent the SQL Injection attacks. Most of techniques are not able to detect the stored procedure attack. To reveal data from database users with valid login id can also enter the malicious queries. In this paper a system is proposed to detect the SQL Injection attacks by using SVM classification and Fisher Score. Proposed System can also classify the users in to normal users or attackers according to the query submitted by them. Weka library is used for SVM classification of the features and for feature selection fisher Score is used. Accuracy for the system is approximately 94%.*

Key Words: SQL Injection attacks, Fisher Score, SVM Classification, Intrusion Detection, and Cyber Security.

1. INTRODUCTION

In the era of the digital revolution the dependency of humans on the computers and robots is increased exponentially also due to the smart phones the dependence upon the internet application is also increased. The Fourth Industrial Revolution is the proof of the use of advanced technology in the world. Computer Systems are under different attacks to steal the valuable information. With the boosting use of internet, use of the web applications has also increased. Most of the web applications have three layer architecture, those are: Presentation layer, Common Gateway Interface (CGI) layer and Database layer as it is the basic architecture for application. The dynamic web applications needs to be always available to all the users, clients, employees, and partners all around the world. As the development of information technology is increasing effectively, these web applications can be accessed from anywhere. By combining different types of attacks most of the web applications have different vulnerabilities in them, due to which they are possible to hack. These attacks include SQL injection attacks, cross site scripting (XSS) attacks. These days the, many web sites are being hacked by attacker by using threat for web application security is SQL Injection Attacks. Structured Query Language (SQL) is used to retrieve the relevant information from the database. SQL injection deals with injection of code into the normal queries to temper the original use of the query. Using combination of SQL injection attacks one can steal important information such as internet banking passwords, mobile banking passwords, ATM pins, user credentials from web applications and even can delete tables from the database. With the development of different applications, massive amount of sensitive personal information of users is been collected in databases continuously [1] [2]. This data can be considered as the most valuable assets of organizations. The number of attempts to hack the valuable data also increases.

Basically Vulnerabilities for web application are [3]:

At the time of coding of the web applications, lack of knowledge for security measures by the developers. Delay in the testing or analysis of the application till the runtime phase. The type specification is not handled properly and usage of the string and number is not defined properly. The input validation of the user is not well defined. Inputs are not checked correctly, very less restriction on the input provided for user. Secure Socket layer can't detect the SQL Injection attacks as it only deals with the certificates and the encryption. Sometimes legal users reveal their own passwords to other attackers. If the application is not developed properly then it can be attacked by SQL Injection attacks.

The Database intrusion attacks, there can be two types [9], depending on the usage of the database. In the first type, users with valid user id and password directly access the database and steal the information. In the other type, attackers indirectly access the database using the vulnerabilities present in the database-driven web applications by using combination of attacks. That is, attackers by altering the original SQL statements attack

the database through the user input values. Different authors have published survey or taxonomy for SQL Injection attacks detection and prevention in [3] [4] [5] [6].

SQL injection attacks are very common type of attacks used to attack the web application [2]. The main types of queries used for attacks are: Tautologies are used for authentication bypass, Logically Incorrect Queries, and Union Query can be used to gain useful information from the database, Stored Procedure is used for storing the malicious query in the database, Piggy-Backed Queries can be used to bombard many queries at a time making server very busy, Inference, and Alternate Encoding are used to Bypass the validation techniques.

SVM Classification:

The support vector machine (SVM) is a training algorithm used to train different types of dataset which can be classified in two or more classes. It can train the classifier in order to predict the class of samples. SVM is based on the concept of decision planes that defines decision boundary and points that form the decision boundary between the classes called support vector are treated as parameters. SVM is based on the machine learning algorithm invented by Vapnik in 1960's [7]. The two key implementations of SVM technique are mathematical programming (primal and dual) and kernel function trick. The main aim of SVM is to find an optimal hyper plane between data point of different classes in a high dimensional space.

(SMO) Sequential Minimal Optimization is useful for solving mathematical programming problems which arises during the training of support vector machines [8]. It is widely used for training support vector machines. System is implemented by importing the weka library. To make classification system uses liner kernel trick.

The rest of the paper is organized as: the next section describes the brief Work Related to SQL Injection. Section 3 describes the Proposed System for Detection of user SQL attacks from users and the results for the system. Section 4 gives the conclusion and future scope of the paper.

2. LITERATURE SURVEY

SQL Injection attack detection: Detection of the stored procedure attacks is discussed in paper [9]. In this paper by using machine learning approach they have detected the attacks at the database level. This approach is useful to detect the stored procedure attacks by using query trees of the SQL query and various kernel functions. SQL queries at the run time stored as tree format are collected from the log files of the postgres SQL database. Here query trees of the SQL queries are transferred to multidimensional array by using feature extraction and transformation. By comparing these multidimensional feature matrices, the normal and malicious queries are classified with the help of the SVM classifier and kernel functions.

Some latest approaches for the countering SQL Injection attack are:

By combining different software diversification in various components of the application, it can be more secured web application [10]. Massive reuse of code on server side and client side has lead to monoculture in designing of web applications. The monoculture is expanding the risk of attacks, in which if hackers are able attack one site or application then they can attack any websites by using same approach. Using multitier diversified software and different combination of layers in web application and avoiding monoculture one can prevent SQL Injection attacks and other different types of attacks.

Secure delivery networks are used in [11], that consists of four types of server's those are DNS server, edge server, parent server, and origin server. Various controls like network layer control, adaptive rate control, application layer control, client reputation rules, at different server's web sites can be used to prevent attacks, without any performance degradation.

SQLRand is a practical protection mechanism against SQL injection attacks [12]. It prevents SQL injection attacks specific to a particular CGI application by using a randomized SQL query language. It uses the concept of instruction-set randomization for SQL, and creates instances of the language that are unpredictable to the attacker. The database parser is used to caught and terminate queries injected by the attacker. If any keywords without randomization found that is a SQL injection. In this way attacker is not able to perform SQL Injection Attacks without the secret key for randomization. By using a proxy for the de-randomization process, SQLRand achieve portability and security.

In Machine learning-based testing approach [13] is used to detect SQL injection vulnerabilities in firewalls. In this they used context-free grammar, RAN (Random generation), ML-Driven for detection of SQLi attacks.

Amenesia [14] is a toolset for preventing SQL Injection Attacks. This toolset consist of Instrumentation module, Analysis module, Runtime monitoring module. Also it has two parts. In its static part, toolset program analysis used to automatically build a model of the appropriate queries that could be generated by the application. In the dynamic part, toolset programs monitor the dynamically generated queries at runtime and then checks it for compliance with the statically-generated model. In run time monitoring Malicious Queries that violate the model can be SQL Injection Attacks and are prevented from executing on the database and reported.

Anamika Joshi and Geetha in [15] have used a classifier which uses combination of Role Based Access Control mechanism and Naïve Bayes machine learning algorithm for detection of SQL Injection attacks. With respect to each blank space in the query they have used the blank separation method and tokenizing method to extract terms, also proposed Role Based Access Control mechanism for detection of attacks. In this way they tried to overcome the limitations in the tool Amnesia to detect SQL Injection.

CANDID is a short form for CANdidate evaluation for Discovering Intent Dynamically [16]. It is the dynamic approach used for mining the structures of the programme intended queries. A formal basis for this dynamic approach is using the notion of symbolic queries. There are two mechanisms, one based on automated byte code transformation and another based on a modified virtual machine (VM), that implement dynamic approach for Java programs to defend against SQL injection attacks. For automatic prevention of the SQL Injection attacks it is a Dynamic Candidate Evaluations method. This framework can be used to solve the issue of manually modifying the applications for creating the prepared statements.

User behaviour detection by using SVM:

Classification of the different Users can be useful for the analysis of the system uses. Until now different authors have published paper for the user behaviour detection System or intrusion system by using the SVM classification. They have used different technique to detect the attack. For intrusion detection system development different authors have used feature extraction such as Review, Syntactical, lexical, and stylistic, n-gram, Stylometric, Behavioral features combined with the bigram, Text features but query tree features are rarely used. Supervised learning is the most frequently used machine learning approach for performing spam detection [17].

3. PROPOSED SYSTEM

3.1 Query tree collection:

Query tree can be used to prevent the SQL injection attacks [18]. System uses the Postgres SQL database to generate the query tree of the SQL statements. The query tree can be obtained the query trees by using following commands in the psql command line interpreter:

```
SET debug_print_plan = on;
SET client_min_messages = debug1;
```

The query trees get stored in the log files of the postgres SQL database. The query and its tree can be manually inserted by the user. To reduce the time the dataset has been generated for the normal query trees and the malicious query trees.

3.2 Feature Extraction and Vector Generation:

Feature extraction is important step for accurate results. Feature extraction module extracts features from the query tree. The difference between the normal query tree and the malicious query tree can be measured by selection of specific features from the query tree. Feature vector is generated from these extracted features [19] [20]. The dataset for the malicious and normal queries feature vectors has been created.
For future Selection the standard deviation and the mean in Fisher Score [21] are calculated as:

$$F(x^j) = \frac{\sum_{k=1}^{c} n_k(\mu_k^j - \mu_j)^2}{(\sigma^j)^2}$$

$$\text{Where } (\sigma^j)^2 = \sum_{k=1}^{c} n_k(\sigma_k^j)^2$$

Feature selection can be used to avoid the redundant features.
Selected features can be used to train the SVM.

3.3 *Training of Vectors:*

The vectors are trained and are stored using the attribute relation file format. These trained features are used for detection of malicious and normal queries. The Weka library consists of source codes which can be imported to perform various tasks. 10-fold class validation is performed which gives (TPRate) true positive rate, (FPRate) false positive rate, accuracy, and the area under (ROC) receiver operating characteristics.

3.4 *Detection by using SVM Classification:*

The submitted query and the trained query are compared by using the vectors in the train.arff and test.arff and the results are shown. SVM classification is done by using SMO and linear kernel [22]. The features of entered query and query tree are compared with the trained dataset.
Generating alert and marking user as malicious user:
As the user enters query to retrieve the data from the database the query tree is compared with the trained dataset by using SVM and the user is marked as malicious user if the submitted query is Malicious query. A list for the malicious users is kept in the database with the query entered and time stamp.

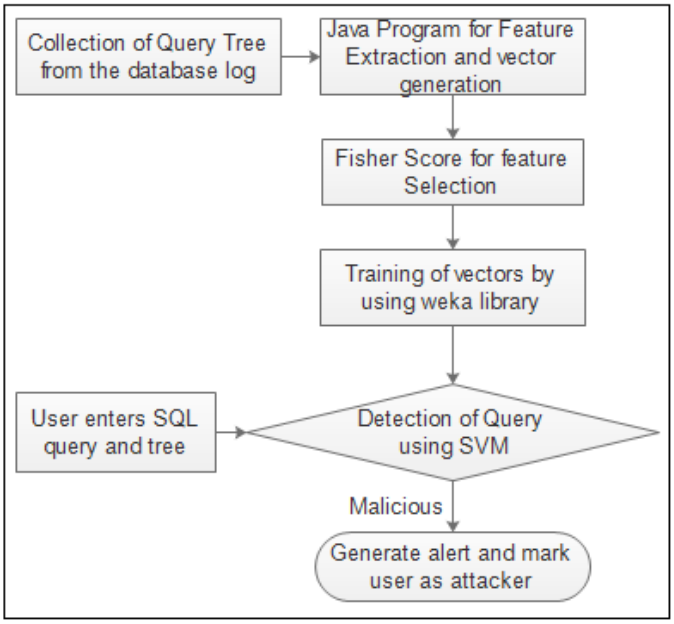

Fig1: System Flow

To demonstrate the experiment an application as movie recommender developed. As dataset system uses Movie lens dataset which contains Users and movies information.. The data contain user ratings given to different movies, more than 1600 movie information, and simple demographic information (age, gender, occupation, zip) for 943 user's [23], also the dataset for malicious query and normal query trees have been generated. Users can search movie, rate movie and insert query to retrieve results from the database. This query can also be detected as normal or malicious queries Users are marked as malicious or attackers if the query is found malicious. Details of the attacker are stored in the database with the time stamp and the query entered.

3.5 *Results and Tables:*

The proposed system can detect the user behaviour by using query tree and the SVM classification. If the query entered by user is malicious then the user is classified as malicious user and alert is generated to the admin. Also the data for the query and the user is saved in database for future use. User is malicious then the user is classified as malicious user entered query and the time stamp is saved.

Table1: Evaluation

Correctly classified instances	94.1176%
Incorrectly classified instances	5.88224%
Kappa Statistic	0.8661
Mean absolute error	0.0588
Root mean squared error	0.2425
Relative absolute error	12.7517
Root relative squared error	50.7253

Table2: Measurements

Measurement	**Normal Class**	**Malicious Class**	**Averaged weight**
TP Rate	1	0.833	0.941
FP Rate	0.167	0	0.108
Precision	0.917	1	0.949
Recall	1	0.833	0.941
F-measure	0.957	0.909	0.94
ROC Area	0.917	0.917	0.917

In this way system can classify the users by using the query submitted by them. Comparison of the query is done by using SVM in WEKA library with the help of the train and test dataset of queries. System has been tested for application based queries of normal and malicious query tree. For total number of different instances 18, accuracy of the system to for classification is 94.117644706%.

4. CONCLUSION AND FUTURE SCOPE

System is Proposed for the SQL Injection attack detection and user behaviour detection by using query tree. The proposed system is tested with sample dataset having various queries. To reduce the redundant features the system uses the Fisher Score for feature selection. The result gained shows that the redundant features are removed. The proposed system is able to detect the malicious query and hence classify the users. In future this type of system can also used for classification of botnet attacks and different security systems by comparing the differences between the queries submitted by the normal user and the robots, also for different type attacks such as cross site scripting attack, low rate attacks in DDOs.

ACKNOWLEDGEMENT

Our sincere thanks go to D. Y. Patil College of Engineering, for providing a strong platform to develop our skill, capabilities and caliber. We would like to thanks all those who directly or indirectly help us in presenting this paper. We hereby take this opportunity to express our heartfelt gratitude towards the people whose help is very useful to complete our project successfully. I would like to express our heartfelt thanks to my guide whose guidance became very valuable for me.

REFERENCES

[1] Jose Fonseca, Nuno Seixas, Marco Vieira, and Henrique Madeira, "Analysis of Field Data on Web Security Vulnerabilities", IEEE transactions on dependable and secure computing, vol. 11, no. 2, march/april 2014.

[2] https://www.owasp.org/index.php/Top_10_2013-Top_10.

[3] Diallo Abdoulaye Kindy and Al-Sakib Khan Pathan, "A survey on SQL injection: vulnerabilities, attacks, and prevention techniques",. 2011 IEEE 15th International Symposium on Consumer Electronics.

[4] Lwin Khin Shar and Hee Beng Kuan Tan, "Defeating SQL Injection", 2013 Published by the IEEE Computer Society.

[5] Amirmohammad Sadeghian, Mazdak Zamani, Azizah Abd. Manaf,"A Taxonomy of SQL Injection Detection and Prevention Techniquest", International Conference on Informatics and Creative Multimedia 2013 IEEE.

[6] Aniruddh Ladole, D. A. Phalke, "A Survey on SQL Injection Attack Countermeasures Techniques", November 2015, International Journal of Science and Research volume 4 issue 11.

[7] Vladimir Vapnik, Steven E Golowich, Alex Smola, "Support vector method for function approximation, regression estimation, and signal processing", 1996, Advances in neural information processing systems 9.

[8] Keerthi, S. Sathiya and Shevade, Shirish Krishnaj and Bhattacharyya, Chiranjib and Murthy, Karuturi Radha Krishna "Improvements to Platt's SMO algorithm for SVM classifier design", Neural Computation,vol 13, 2001, MIT Press.

[9] Dong Hoon Lee, Mi-Yeon Kim, "Data-mining based sql injection attack detection using internal query trees" ,expert systems with applications 41 (2014) 5416 to 5430.

[10] Simon Allier, Olivier Barais, Benoit Baudry, Johann Bourcier, Erwan Daubert, Franck Fleurey, Martin Monperrus, Hui Song, Maxime Tricoire, "Multitier Diversification in Web-Based Software Applications", January/February 2015, IEEE Software.

[11] David Gillman, Yin Lin, Bruce Maggs, Ramesh K. Sitaraman, "Protecting Websites from Attack with Secure Delivery Networks", IEEE 2015.

[12] S. W. Boyd and A. D. Keromytis. SQLrand: Preventing SQL Injection Attacks. In Proceedings of the 2nd Applied Cryptography and Network Security (ACNS) Conference, pages 292–302, June 2004.

[13] Dennis Appelt, Cu D. Nguyen, Lionel Briand, "Behind an Application Firewall, Are We Safe from SQL Injection Attacks?", 2015, IEEE.

[14] William G.J. Halfond and Alessandro Orso , "AMNESIA: Analysis and Monitoring for NEutralizing SQLInjection Attacks", November 2005,ACM.

[15] Anamika Joshi, Geetha V, "SQL Injection Detection using Machine Learning", 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies.

[16] Prithvi Bisht, P. Madhusudhan, V. N. Venkatakrishnan "CANDID: Dynamic Candidate Evaluations for Automatic Prevention of SQL Injection Attacks", 2010, ACM.

[17] Crawford, Michael, et al. "Survey of review spam detection using machine learning techniques." Journal Of Big Data 2.1, Springer (2015): 23

[18] Buehrer, G., Weide, B. W., & Sivilotti, P. A. (2005) Using parse tree validation to prevent SQL injection attacks. In The fifth international workshop on software engineering and middleware (pp. 106–113). ACM.

[19] Yi Ru, Alina Campan, James Walden, Irina Vorobyeva, Justin Shelton, "An effective log mining approach for database intrusion detection", Systems Man and Cybernetics (SMC)2010 IEEE.

[20] Peter Buneman Byron Choi Wenfei Fan Robert Hutchison Robert Mann Stratis D. Viglas, "Vectorizing and Querying Large XML Repositories", International Conference on Data Engineering, 2005 IEEE.

[21] Quanquan Gu, Zhenhui Li, Jiawei Han, "Generalized Fisher Score for Feature Selection", journal ArXiv e-prints 2012.

[22] Mark Hall, Eibe Frank, GeoffreyHolmes, Bernhard Pfahringer, Peter Reutemann, Ian H Witten,. "Data mining: Practical machine learning tools and techniques". Elsevier (2011).

[23] Riedl, J., & Konstan, J. (1998). Movielens dataset. <http://grouplens.org/>. Dataset.

BIOGRAPHIES

Aniruddh Ladole pursuing Master's in Computer Engineering from DY Patil College of Engineering. His area of interest is Cyber Security, Database Security, Intrusion detection system.

Mrs D. A. Phalke completed her Master's in Computer Engineering from D Y Patil College of Engineering and pursuing PhD in Computer Science and Information Technology at Department of Technology, Savitribai Phule Pune University. She is having 14 years of experience of PG and UG teaching and 13 publications in International Journals and 16 in International Conference. Her area of interest is Data Mining, Database Security and Multimedia Data.