International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | Impact Factor: 7.488 |

|| Volume 8, Issue 5, May 2020 ||

Detection and Prevention of SQL Injection Attacks Using Hybrid Approach

Madhur Thakral[1], Dr. Paramjeet Singh[2], Dr.Shaveta Rani[3]

Student, Dept. of CSE, GZSCCET MRSPTU, Bathinda, Punjab, India[1]

Professor, Dept. of CSE, GZSCCET MRSPTU, Bathinda, Punjab, India[2]

Professor, Dept. of CSE, GZSCCET MRSPTU, Bathinda, Punjab, India[3]

ABSTRACT: SQL injection attacks are the most hazardous attacks in the context of web applications as they can go around the system verification and can remove the private information of the user of the appropriate web application. There are many types of sql injection attacks like first-order attack, second-order attack, tautology attack, piggy-back attack, stored-procedures etc. A system should be implemented to monitor and avoid these types of attacks system should be developed that will monitor and prevent these attacks. The proposed system is combination of two approaches. One approach is applied to prevent system at run time and other is applied to prevent system at compile time. This hybrid approach is implemented and the proposed system using this approach shows 95.12% accuracy for various types of inputs.

I.INTRODUCTION

Sql injection attacks exploit the vulnerability of web applications so as to retrieve and modify the sensitive information present in the web application database. Web applications which are mostly used such as online shopping applications, online banking applications, web-portals are frequently harmed and accessed by malicious hackers using sql injection. Sql injection uses timing delays, input performance and alike parameters to identify which system vulnerabilities can be exploited. Hybrid approach is developed to secure web applications.

The hybrid approach performs Static Analysis to prevent system from sql attacks at compile time and Dynamic Analysis to prevent system from sql attacks at run time.

1. 1 During Static Analysis, the input queries are compared with queries present in meta string libraries at compile time. Keyword matching is done to prevent the system from sql attacks. It is also named as Positive Tanting.

1.2 During Dynamic Analysis the input queries are compared with the queries present in primary list at run time. Machine learning is implemented to prevent system from sql attacks. It is also named as Negative tanting.

This approach mainly involves Application tier, Presentation tier and Database tier to detect and prevent sql injection attacks. Many types of sql injection attacks are monitored and prevented.

Types of SQL Injection Attacks are:

First Order Attack: It usually occurs when a attacker interacts with a web application and receives the desired result from the application immediately.

Second Order Attack: Attacker inserts malicious inputs to the database and indirectly triggers a SQLI which is used at a later time.

Tautology Attack: Conditional statements are injected in such a way that result evaluates to be always true. It is a form of SQLI manipulation.

Error Based SQL Injection: This technique is used to quickly exploit the database of an application. The intention of this attack is that the sensitive information of different databases can be stored into error messages in terms of receiving sql expression.

Union Query Attack: In this method attacker inserts an injected query along with a safe query with the help of union operator. After the execution of safe query, the system executes the injected query which results in unauthorized access of database.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | Impact Factor: 7.488 |

|| Volume 8, Issue 5, May 2020 ||

Stored Procedures: in this attack, attacker tries to execute the stored procedures already presents in the database.

II.LITERATURE SURVEY

Atefeh Tajpour,Suhaimi Ibrahim, & Mohammad Sharifi [2] described that sql injection attacks are the attacks in which sql code is injected to web form to gain access to web application and manipulates its sensitive information. This is done to affect the confidentiality and integrity of web application.

Jalal Omer Atoum and Amer Jibril Qaralleh
described that access to the data should be well controlled by assigning the access rights to the users. For this Static and Runtime analysis should be implemented.

Shaul, J., and Ingram developed a system to ensure the security of sensitive information. Even though the database is secured from being hacked, sensitive data should be encrypted in the database or through thenetwork.

W. G. J. Halfond *et al* This paper introduced an approach that uses trusted strings to create sensitive parts of the SQL query strings. All the attacks can be stopped without generation of false positives. It is used to improve the efficiency by reducing the amount of required information.

III.PROPOSED METHODOLOGY

In this stage, programming configuration is set up from the necessities. Framework configuration is useful in indicating equipment and programming necessities and furthermore helps in characterizing the general framework design. Proposed Methodology is based on Hybrid Approach. This hybrid approach will use both positive and negative tanting and tries to prevent SQLI at runtime and compile time. Algorithm for the proposed system consists of two phase: -

Phase 1: This phase checks the input query at compile time. System will check for various SQLI attacks at compile time during this phase. If any sqli attack is found in the input query system will block the input query at compile time. This query is neither compiled nor executed. If input query passes the first phase after checking then it will enter into second phase.

Phase 2: This phase will check sqli at run time. It is further divided into two phasesFirst phase is called training phase. Second phase is known as SQLI Detection phase.

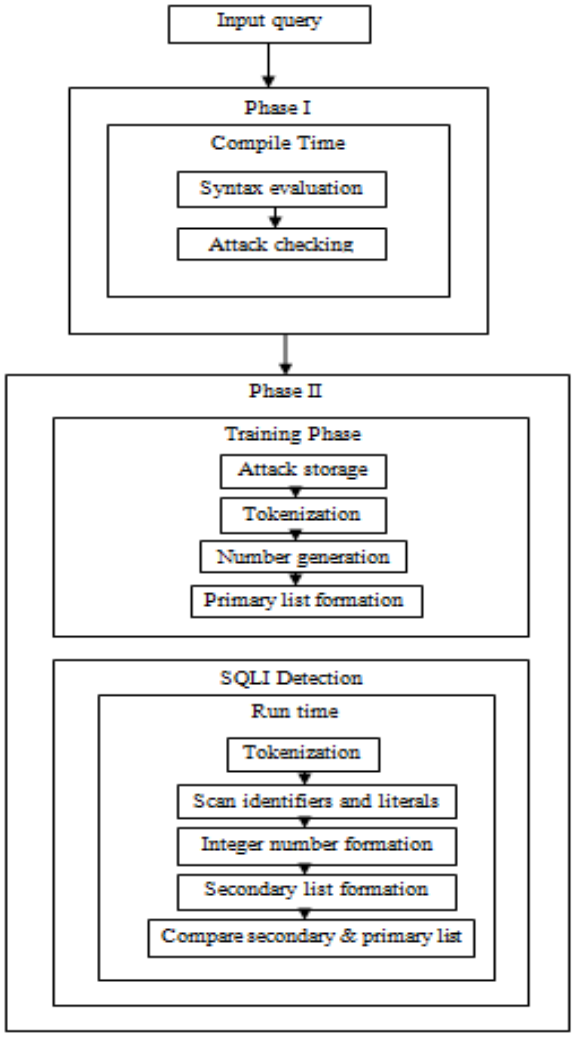
Training phase: - In this phase, the database administrator will store all the attacks in the database. Attacks are divided into tokens and these tokens are converted into integer numbers with the help of ASCII values according to positions. Every attack is represented by an integer number. A primary list of attacks is formed to detect SQLI in input queries.

SQLI Detection phase: - In this phase, input queries are tokenized and converted into integer values and a secondary list is formed. This list is compared with primary list to detect SQLI attacks. If SQLI is found , it is blocked otherwise input query is compiled and executed. In this way SQL injection attacks are monitored and prevented in web applications.
A flow chart is designed to implement the technique for the proposed hybrid approach. It clearly specifies the steps to detect and prevent sql injection attacks at compile time as well as at run time.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | Impact Factor: 7.488 |

|| Volume 8, Issue 5, May 2020 ||

Flow chart of proposed system



IV.RESULTS

Proposed as well as existing system are implemented using .NET VISUAL STUDIO as a simulator and different databases such as MY SQL, MY ACCESS AND SQL SERVER as backend. A hybrid approach is used by proposed system to detect and prevent different types of SQL injection attacks such as Tautology attack, Union queries, Piggy-back attack, Error-injection attack, Timing attack, Inference attack etc. Parameter under consideration is Accuracy. Proposed system is checked for different types of queries and it shows 95.12% accuracy. The efficiency of proposed system is checked for more than200 queries and it prevents them very efficiently.

Parameter	Value
Total no. Of Inputs tested	250
Total number SQLIA Handled	230
No. Of Various types of Attacks	12
System Accuracy	95.12%

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | Impact Factor: 7.488 |

|| Volume 8, Issue 5, May 2020 ||

Proposed system shows accuracy as compared to existing techniques such as: -

CANDID: - Dynamic Candidate Evaluation technique is a method to prevent SQL injection attacks by inferring intended queries by considering the symbolic query computed on a program run. It retrofits web applications written in java to defend them against SQL injection attacks.

DYNAMIC TANTING:- It is based on tanting input strings, tants are tracked along the run of a program and check the keywords of a query, if they are already tants with some keywords of another query. It is a powerful form of SQL injection prevention technique.

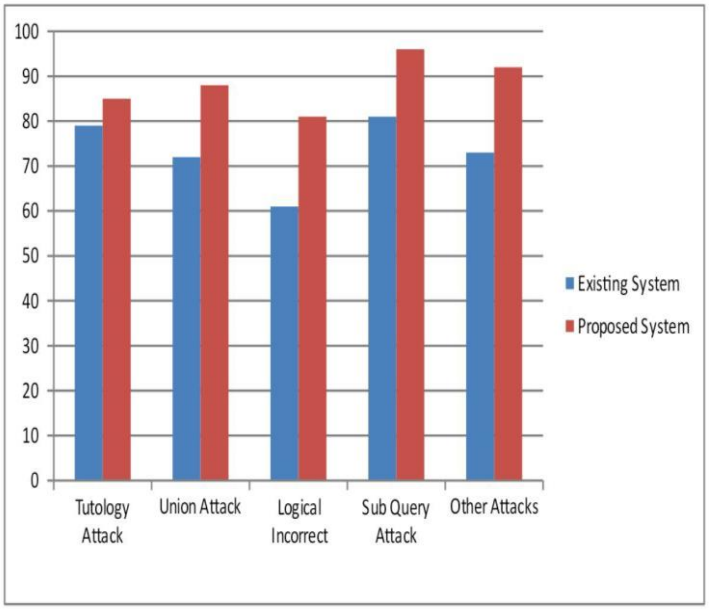
AMNESIA: - Analysis for monitoring and Neutralizing SQL injection attacks is a tool for securing and protecting web applications. A model- based approach is used by this technique that combines static analysis and run time monitoring. It is a highly effective technique.

SQL Rand:- It is a technique to prevent SQL injection attacks. In this technique, queries inserted by the attacker are caught and then terminated. Database parser is used to terminate these input queries

Schemes	Tautology	Logically Incorrect Queries	Union Queries	Stored Procedure	Piggy Backed Queries	Inference	Alternate Encoding	Compile Time Attack Checking	Multi-threading Technology Used	Overall Attack Immunity
AMENSIA	Yes	Yes	Yes	No	Yes	Yes	Yes	No	No	66.66%
SQLrand	Yes	No	Yes	No	Yes	Yes	No	No	No	44.44%
CANDID	Yes	No	No	No	No	No	No	No	No	11.11%
SQLguard	Yes	No	No	No	No	No	No	No	No	11.11%
SQLIPA	Yes	Yes	Yes	No	Yes	Yes	Yes	No	No	66.66%
Negative Tainting	Yes	Yes	Yes	No	Yes	Yes	Yes	No	No	77.78%
Proposed System	Yes	Yes	Yes	Partially	Yes	Yes	Yes	Yes	Yes	95.12%

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | Impact Factor: 7.488 |

|| Volume 8, Issue 5, May 2020 ||



Comparison of Existing system and Proposed system.

Comparison of proposed and existing techniques based on different types of SQL Injection attacks.

V.CONCLUSION AND FUTURE WORK

CONCLUSION

The proposed work presented a robust system for the detection and prevention of SQL injection attacks for web applications. As SQL injection attacks are the most dangerous threats for a web application as they can bypass the user authentication and can also steal the information and important data of the web user.

FUTURE WORK

In the future, the proposed system can be improved by implementing multithreading approaches along with the existing techniques to increase the attack detection speed. Also, it can be implemented using oracle database.

REFERENCES

[1] Dr. Ahmed Ghafarian, "A Hybrid Method for Detection and Prevention of SQL Injection Attacks," in IEEE Computing Conference 18-20 July ,2017. pp. 835-838.

[2] Atefeh Tajpour, Suhaimi Ibrahim, & Mohammad Sharifi,"Web Application Security by SQL Injection Detection Tools". IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 2, No 3, March 2012, ISSN (Online): 1694-0814.Bandhakavi, S., Bisht, P., Madhusudan, P., and Venkatakrishnan V., "CANDID: Preventing sql injection attacks using dynamic candidate evaluations", in the Proceedings of the 14th ACM Conference on Computer and Communications Security, 2007

[3]Jalal Omer Atoum, Amer Jibril Qaralleh, Princess Sumaya University for Technology, Amman and Jordan, "International Journal of Database Management Systems" (IJDMS) Vol.6, No.1, February 2014.

[4] Shaul, J., and Ingram A., " Practical Oracle Security, Rockland: Syngress Publishing, Rockland, MA: Syngress Pub"., c2007.

[5] W. G. J. Halfond, A. Orso, and P. Manolios, "Using positive tainting and syntax-aware evaluation to counter SQL injection attacks," in *Proceedings of the 14th ACM SIGSOFT international symposium on Foundations of software engineering - SIGSOFT '06/FSE-14*, 2006, pp. 175–185.

[6] Prithvi Bisht, P.Madhu sudan, V.N. Venkatakrishnan, University of Illinois, Chicago, "Dynamic candidate evaluation for automatic prevention of SQl injection attacks" in, ACM transactions of Information and System Security,Volume 13 Issue 2,February 2010.

[7] Sruti Bandhakavi, Prithvi Bisht, P.Madhu sudan, V.N.Venkatakrishnan, University of Illinois, Chicago. USA, "Preventing SQL Injection Attacks using dynamic candidate evaluation". Proceedings of the 14 ACM conference on computer and communication security, pp 12-24 ,2007.