ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 3, March 2019

A Survey on SQL Injection Attacks and Methods to Detect and Prevent Them

Khushboo Choubey [1], Prof. Priyanka Jain [2]

M Tech Student, Department of Computer Engineering, GNCSGI, Jabalpur, India[1]

Associate Professor, Department of Computer Engineering, GNCSGI, Jabalpur, India[2]

ABSTRACT:-SQL injection attacks occur when an unauthorised user gets an opportunity over web applications.SQL injection attack is most commonly attack used by the attackers to implement their goals. In SQL injection attack one can take advantage of poorly coded SQL queries to execute the unauthorised websites .By this that user can also access confidential data of any organisation. For doing this attack that one target a vulnerability of that particular web application for an instance edit the SQL query for their significance. By this attack they exploit the security of DB layer of an application.
This survey paper presents an effective survey on SQL injection attack and a method to detect and prevent them.

KEYWORDS- SQLIA, unauthorised user, web security, confidential, SQL string, authentication, DB layer.

I. INTRODUCTION

Today in web applications data management, application processing, and presentation becomes an important task for the web site developer. Instead of writing the entire application, now days the developers have to edit only a few codes, which help in designing, maintaining and protecting the data. The first phase which is known as data management phase consists of a database server, where secret or confidential information related to that application and the user is saved and fetched. The data from this phase is safely fetched by the authenticated user, for storing their record and their relationship, and for presenting the data in the created web page.

In general, the SQL queries are processed by the SQL query processor and get executed. The resultant of this procedure is returned to the application server. The application server fetches the returned data and after checking it takes a decision to submit the data in the dynamic web page. The journey of static to dynamic web pages provides an opportunity to use of database in web applications. Due to the scarcity of secure coding techniques, SQL injectors prevail in a large set database server for execution. The input parameters which are provided by the user may or may not be trustworthy. Query processor will execute the query and without considering its type send the resultant to the user. But the query can still contain logically incorrect codes which are malevolent or malicious. The attackers take advantage of such codes and can provide this malicious code in the input parameter. If there is no proper separation between the program instructions and the data provided by the user the malicious code by the attacker may get executed.

By just modifying some codes of the SQL query, the attacker may detract confidential information from the database and may get control over the database and the database server. This technique of getting control over web application is very popular among the attackers by the name of “SQL injection attack.” The additional advantage for the attacker is that it uses port 80 (a default port for HTTP) to communicate, and this port always remains open and neither blocked nor filtered by the firewall. In this survey paper SQL injection attack and the steps to detect and prevent the data from the attacker have been elaborated.

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 3, March 2019

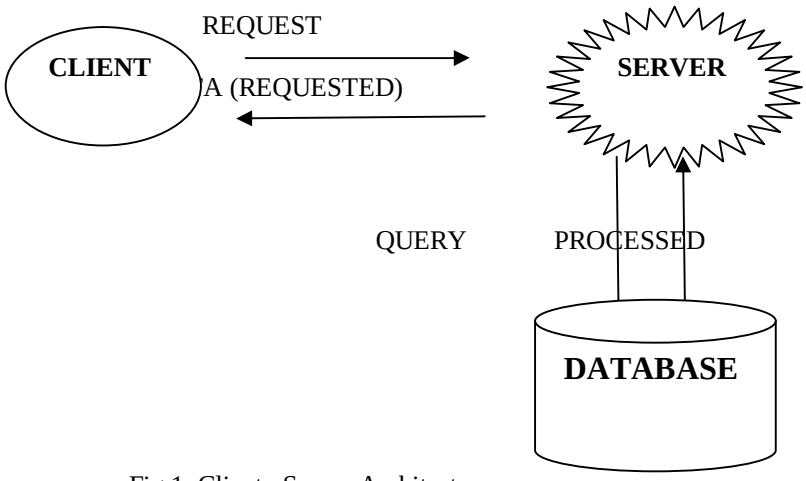

Fig.1. Client –Server Architecture

Fig.1.shows how client and server site communicate with each other in networking area. The client requested the data needed .The server fetches the request from the client and through the query string peek their own database, i.e. they have the requested data or not .If they don't have ,the server reply back that they don't have the data requested by the client or if they have send the date requested.

II. RELATED WORK

1. Sonewar and Thosar explained the methods used for gathering information to launch SQLIA. In this the authors explained the exploiting vulnerabilities in web application using payload method.

2. A dynamic technique of query matching which quickly detects the attack was proposed by Raja P.K .In his work he minimizes the processing time by using dynamic query matching process.

3. A static analysis tool was discussed by Xiang Fu. Kai Qian and Lixin Tao. In their work they discussed a tool which uses hybrid constraint solver at hot spot to inspect the submitted query and identify the SQLIA.

4. Pandurang and Karia proposed a mapping modal used to detect and prevent SQL injection attacks. In their modal they restrict the damage caused by the attackers by making each client containing their own container.

5.Chenyu and Fan proposed an intention oriented approach on SQLIA detection process .They uses deterministic finite automat to represent SQL string and for checking the vulnerability in that SQL string

III. TYPES OF SQL INJECTION ATTACK

1. Tautologies:

Tautology-based attacks work by injecting one or more conditional SQL statement queries in such a way that make the SQL command rate as a true condition such as ('1=1') . This technique is mostly used to bypass authentication on web pages results as to access the database.

For example:

```
SELECT id FROM users WHERE emailid='id' AND password='password' OR 1=1'
```

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 3, March 2019

2. Piggy-backed Query:
In this type of Query attacker tries to add some queries to the original query string. On the successful attack the attacker receives the database and executes a query string that contains different type of multiple queries. In this type of attack the first query is original whereas the additional queries are injected. This attack is very harmful as in this the attacker use it to inject virtually any type of SQL command.
For example:
SELECT * FROM user WHERE id='admin1' AND password='12345'; DROP TABLE user1; --';
Here database hacker add a query string as two query separated by ";" and executes both. The sub query is malicious query and it affects the database to drop the user table in the database.

3. Logically Incorrect:
In logical incorrect attack the attacker takes advantage of the error messages that are flashed in the screen by the database for a wrong query. These error messages contain important information that allow attacker to take out the vulnerable parameter in an application.
For example:
SELECT * FROM user WHERE id='11111' AND password='12345' AND CONVERT (varchar, no);

4. Union query:
This query injection is also called as statement injection attack. In union query attack the attacker insert additional statement into the original SQL string. This attack can be successfully done by adding either a UNION query string or a statement of the form **";< SQL statement >"**. The output of this type of attack is that the database returns a dataset that is the union of the resultant of the main query with the output of the inserted query.
For example:
SELECT * FROM user WHERE id='11111' UNION SELECT * FROM member WHERE id='admin1' --' AND password='12345';

5. Stored Procedure:
This type of attack also known as piggyback attack .In this procedure, the attacker aims on the stored procedures which are presented by the database system. Stored procedures can be run directly by the database engine. Stored procedure is a code and it can be vulnerable as program code. For authorized/unauthorized user/attacker the stored procedure returns true/false.
For example:
SELECT accounts FROM users WHERE login= '11111' AND pass='12345 '; SHUTDOWN;--';

6. Inference:
In this type of attack, the attacker changes the behaviour of a database. There are two well known procedures that are based on inference**: blind injection and timing attacks**.

⇨ **Blind Injection**: Sometimes developers hide the error details which help the user to safe the database by the attacker as by logical incorrect attack. In this situation attacker has to face a generic page provided by developer, except an error message. An attacker can still hack the data by asking a string of True/False questions through SQL codes. Consider two possible injections into the login field.
For example:
SELECT accounts FROM users WHERE userid= '11111' and 1 =0 --AND pass = AND pin=1SELECT accounts FROM users WHERE loginid= 'aaa' and 1 = 1 -- AND pass = AND pin=1
If the statements are secured, both queries would be unsuccessful, because of validation process. But if there is no validation, the attacker got the opportunity.

⇨ **Timing Attacks**: In timing attack, an attacker collect the information from a database by considering timing delays in the database's responses. In this technique the attacker uses if-then statement which causes the SQL engine to execute a

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 3, March 2019

long running query or a time delay SQL string depending on the logic injected. This attack is similar to blind injection and attacker can then count the time the pages taken to load to determine if the injected statement is true..WAITFOR is a type of keyword along the branches, which causes the database to delay its response by a specified time.

For example:

declare @ varchar (800) select @s = db_name1 () if (ascii (substring (@s, 2, 2)) & (power (2, 0))) > 0 waitfor delay '0:0:6'

Database will pause for every six seconds if the first bit of the first byte of the name of the current database is 2. Then query is then injected to generate a delay in response time when the condition is true

IV. DETECTION AND PREVENTION TECHNIQUES AGAINST SQLIA

In this section we present some of the most popular techniques, which are used to either DETECT or PREVENT SQLIA

- **A developer develops a technique which is known as AMNESIA (Analysis and monitoring for neutralizing SQLIA).**

SQLIA while they don"t overcome the modern types of SQLIA [8], as follows:

ϖ In [9] the author developed a technique, which is called AMNESIA, that stands for Analysis and Monitoring AMNESIA is a combination of dynamic and static analysis for detecting and preventing web application vulnerabilities at the runtime. This method uses static analysis to generate different type of queries. In the next dynamic phase, AMNESIA checks all queries before they are sent to the database and validates each of them.

- **MHAPSIA Modal (a modal based on hybrid approach to prevent SQLIA in PHP)**

This is an Integrated Approach to Prevent SQL Injection attack. An author proposed a query model that uses the hybrid approach. A model that is the modification of the existing MHAPSIA model .This is done by incorporating the logical queries to prevent SQL IA and a proposed algorithm to prevent the attack.

- **Runtime Monitors for Tautology Based SQL Injection Attacks**

In this method the author proposed a framework to control tautology based SQLIAs in web applications by using a post deployment monitoring technique. The basic idea behind their proposed method i.e., the Source code contains some critical elements that interact with the external user by accepting their inputs, developing queries, and processing them by accessing the internal database. To control the behaviour of an application during its execution there is a method in which ,when a critical element generate the checkpoint and in resultant follows an invalid path, the runtime monitor immediately detects the abnormal behaviour of the application due to the critical element and notifies the admin about the abnormal response of the application. This model generates minimum false positives.

- **An Authentication modal for Preventing SQL Injection Attack Using Hybrid Encryption (PSQLIA-HBE)**

This method proposed an authentication modal in which they introduce an algorithm that uses both Rivest-Shamir-Adelman (RSA) and Advance Encryption Standard (AES) algorithms to prevent an SQL IA. In this method, a unique secret key is given to every client known as private key. On the server side, the server uses a combination of both a private key and public key for RSA encryption. There are three phases by which the authenticated user are validated, namely, the Registration, Login, and Verification phases. Encryption method is used in login phase .For encrypting the user name and password, symmetric key encryption (SKE) is used with the help of the user's private key. To encrypt the query, this method uses asymmetric key encryption (AKE) by using the server's public key. The proposed method is very useful as it needs 961.88 ms for encryption or decryption, which is very negligible.

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 3, March 2019

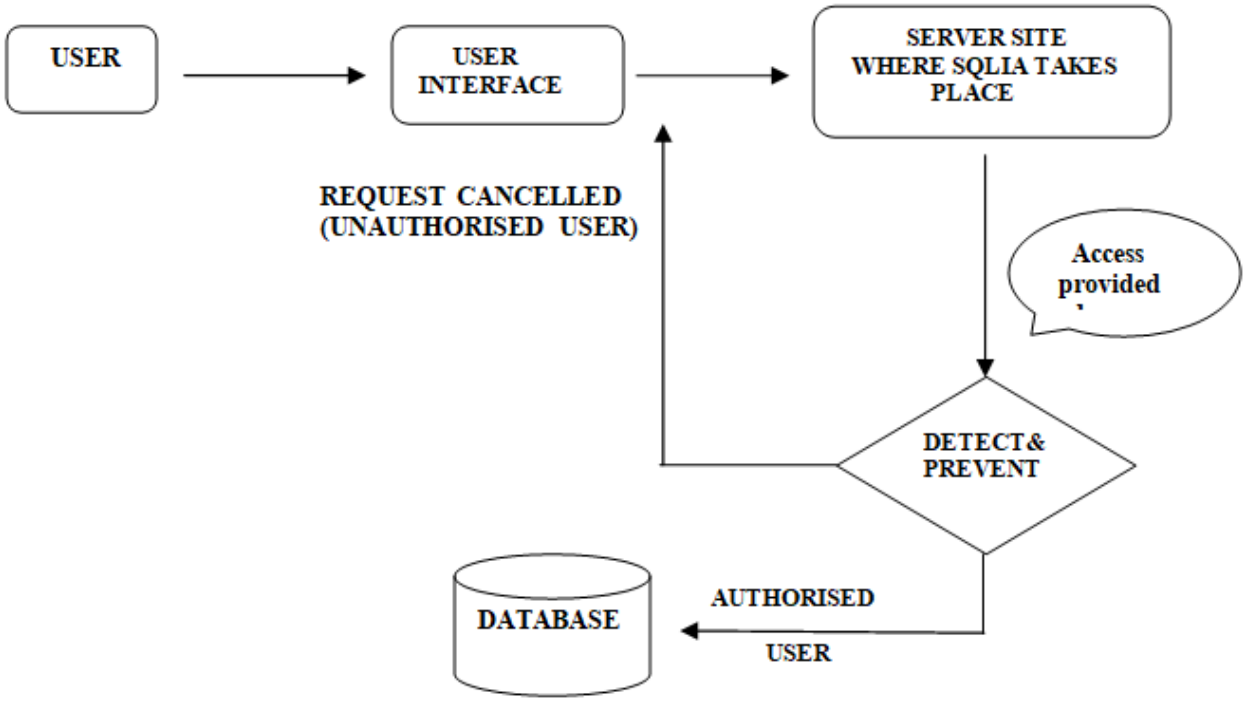

Fig.2. SQLIA WITH ITS DETECTION AND PREVENTION SCHEMA

Fig.2.shows the labelled schema how SQL injection attacks are detected and prevented by their specified method .
In fig.1. we had already defined the usual process takes place when user or client request the needed data, without indentifying that the user is authorised to fetch the data or not .If there is no secure layer between client server architecture the important data get in the wrong hands.
During detection and prevention of SQLIA,when the user requested the needed data after passing through the user interface the request is send to the server site .If there is no attack takes place the only authorised user can fetch the needed data .Now a days due to SQLIA the unauthorised user can also fetch the data and other important and confidential details by just adding some query string or by editing the existing query. We can overcome this unauthorised attempt by testing the string through the SQLIA detection and prevention methods as shown in the above fig.

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 3, March 2019

V. COMPARATIVE ANALYSIS BETWEEN TYPES OF SQLIA

TYPES OF ATTACK	THOSE TECHNIQUES WHICH CAN PREVENT ALL TYPES OF ATTACK	THOSE TECHNIQUES WHICH CAN PARTIALLY PREVENT ALL TYPES OF ATTACK	THOSE TECHNIQUES WHICH ARE NOT ABLE TO PREVENT ALL TYPES OF ATTACK
1.Tautology	43%	57%	0%
2.Piggy-backed query	43%	57%	0%
3. Logically Incorrect	43%	57%	0%
4. Union query	43%	57%	0%
5. Stored Procedure	29%	57%	11%
6. Inference	43%	57%	0%

The above table shows the comparison between different types of SQL injection attacks .It also describes that how different techniques can prevent these types of SQLIA either partially or fully.
The tables simply defining the related comparison between the attacks as how much percentage they are able to prevent the attacks or are not able to do so.

V. RESULT&FUTURE WORK

All types of SQL injection attack were compared and we can get a resultant that all of them are able to prevent SQL injection attacks either fully or partially. We are using a well known technique tautology and defining the process how and by which manner it is detecting the SQL injection attacks and also the method/tool used to prevent that type of attack.
In future, we extend our work by giving an implementation review of tautology method and also we test its performance with the existing methods of SQLIA.

VI. CONCLUSION

SQL injection attack is a common type of attack. This attack becomes a serious security concern for the DBMS service provider over the Internet or over web application. In SQL injection attacks, the attacker can take advantage of vulnerability in Web application. By taking an advantage of this the attacker get an opportunity to steal, edit, delete confidential and critical unauthorised data. The hackers can perform all those operation for which they are not authorised. This paper represents an effective survey on SQL injection attack and their detection, prevention methods.

REFERENCES

1. Alwan ZS & Younis MF, “Detection and Prevention of SQL Injection Attack: A Survey”, International Journal of Computer Science and Mobile Computing, Vol.6, No.8, pp.5–17,(2017).
2. Priyaa BD & Devi MI, “Hybrid SQL injection detection system”, 3rd International Conference on Advanced Computing and Communication Systems (ICACCS), pp.1-5,(2016).

ISSN (Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 3, March 2019

3. Qbea'h M, Alshraideh M & Sabri KE, "Detecting and preventing SQL injection attacks: a formal approach", Cybersecurity and Cyberforensics Conference (CCC), pp.123-129,(2016).
4. Voitovych OP, Yuvkovetskyi OS & Kupershtein LM, "SQL Injection prevention system", International Conference Radio Electronics & Info Communications (UkrMiCo),pp.1-4,(2016).
5. Brynielsson J & Sharma R, "Detectability of low-rate HTTP server DoS attacks using spectral analysis", IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pp.954-961,(2015).
6. Qian L, Zhu Z, Hu J & Liu S, "Research of SQL injection attack and prevention technology", International Conference on Estimation, Detection and Information Fusion (ICEDIF), pp.303-306,(2015).
7. Diksha G. Kumar, MadhumitaChatterjee "Detection Block Model for SQL Injection Attacks" I.J. computer Network and Information Security, 2014
8. BojkenShehu, AleksanderXhuvani "A literature Review and comparaative analysis on SQL injection: Vulnerabiities, attacks and their detection and prevention Techniques" International Journal of Computer Science Issues, Vol 11,Issue 4, no1 2014
9. GeogianaBuja, Dr. Kamarularifin Bin AbdJalil, Dr. Fakariah Bt. HjMohd Ali, The Faradilla Abdul "Detection model for SQL Injection Attack: An approach for preventing a web application from the SQL injection Attack"IEEE Symposium on Computer Applications and Industrial Electronics, April 2014
10. NunoSeixas, Marco Vieira, Jose Fonseca, Henrique Madeira "Analysis of field data on web security vulnerabilities "IEEE Transactions on Dependable and secure computing Vol. 11 No.2 March/Aril 2014
11. HossaianShahriar, Mohammad Zulkernine, "Information Theoretic Detection of SQL Injection Attacks" International Symposium on highAssurance systems Engineering, IEEE 2014
12. Hussein AlNabulsi, IzzatAlsmadi,, Mohammad AlJarrah "Textual Manipulation for SQL Injection attack" I.J. computer Network and Information Security, 2014
13. Monali R. Boradel, Neeta A. Despande "Extensive Review of SQLIA's Detection and Prevention Techniques" International Journal of Emerging Technology and Advanced Engineering ISSN 2250- 2459, ISO 9001:2008 Certified Journal, Volume 3, Issue 10, October 2013
14. Shelly Rohilla, Pradeep Kumar Mittal "Database Security by Preventing SQL Injection Attacks in Stored Procedure" Journal of Advanced Research in Computer Science and software Engineering Volume 3, Issue 11 November 2013.
15. JaskanwalMinhas Raman Kumar "Blocking of SQL Injection attack by Comparing Static and Dynamic queries" International Journal of computer network and Information Security 2013
16. Mihir Gandhi, JwalantBaria "SQL INJECTION Attacks in Web application"International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-6, January 2013
17. Natarajan K & Subramani S, "Generation of SQL-injection free secure algorithm to detect and prevent SQL-injection attacks", Procedia Technology 4 Elsevier Ltd, pp.790–796,(2012).
18. . Fonseca J, Vieira M & Madeira H, "Evaluation of web security mechanisms using vulnerability & attack injection", IEEE Transactions on Dependable and Secure Computing, Vol.11, No.5,pp.440-453,(2012).
19. Djuric Z, "A black-box testing tool for detecting SQL injection vulnerabilities", Second International Conference on Informatics and Applications (ICIA),pp.216-221,(2013).
20. Buja G, Jalil KBA, Ali FBHM & Rahman TFA, "Detection model for SQL injection attack: An approach for preventing a web application from the SQL injection attack", IEEE Symposium on Computer Applications and Industrial Electronics (ISCAIE), (2014), pp.60-64.
21. Pushpa BR, "Enhancing Data Security by Adapting Network Security and Cryptographic Paradigms", International Journal of Computer Science and Information Technologies, Vol.5, pp.1319–1321(2014).