STUDY ON SQL INJECTION-THREATS, ATTACKS, TYPES, PREVENTION TECHNIQUES AND TOOLS

[1]MEHAR SOOD, [2]SMITA SINGH

[1,2]Computer Science and Engineering Department MUST Laxmangarh Sikar Rajasthan India
E-mail: [1]Meharsood321@gmail.com, [2]smi126.nov@gmail.com

Abstract- SQL injection refers to a technique where an attacker can execute malicious SQL statements that control a web application's database. SQL injection vulnerabilities can attack any website or application that makes use of the database server. This vulnerability is one of the oldest and most dangerous of all the web application attack. In SQL injection attack an attacker tries to gain unauthorized access on the database which allows an attacker to spoof the identity, destroys the data present on the system, and changes the records present on the database. The main consequences of SQL injection attack are loss of confidentiality, loss of authentication, loss of authorization and lack of integrity. In this paper we will focus on SQL injection, all the SQL injection attacks, its threats, and the methods used to detect and to prevent SQL injection with the SQL queries.

I. INTRODUCTION

SQL injection is a code injection technique, used to attack data-driven application in which the hacker manipulates the SQL statements to obtain access to the database and its sensitive information. SQL injection is application level security vulnerability. The main intent of the SQL injection attack is to get illegal access to the database, to extract the information, to modify the records present in the database, to delete the records from the database. Basically, SQL injection attack is all about unauthorized access to the database system. Any website or any web application that make use of the database is vulnerable to SQL injection attack.

SQL injection is considered among the top 10 web application vulnerabilities. The main reason for SQL injection vulnerabilities is insufficient validation of inputs at the application layer.

The major consequences of SQL injection attacks are [1]:

- Loss of confidentiality: A hacker can obtain access to the database and other sensitive information.
- Loss of authentication: Without providing authentic username and password a hacker can access the database.
- Lack of data integrity: This refers to the accuracy and consistency of the data in the database.
- Loss of authorization: An attacker can leak complete information present on the system.

According to the statistics recorded in 2012, September shows that SQL injection is one of the most dangerous attacks.is one of the most dangerous attacks.

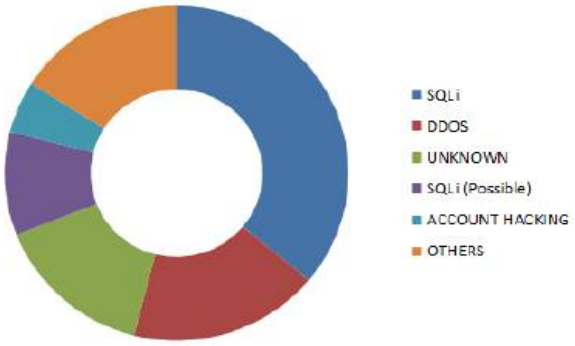

Fig.1. Statistics of various attacks

To address SQL injection attack various solutions are proposed such as input validation at client side, limiting the input size, usage of special characters as they are rarely used by the users in the data, etc.

The combination of various proposed approaches is used to prevent and to detect SQL injection attacks.

II. SQL INJECTION

SQL injection is a language that allows a hacker to obtain illegal access to the web application's database and other sensitive information. A hacker tries to gain access to the database by manipulating SQL injection commands and this allows hacker to update or to delete the data stored in the database. SQL injection can be defined as the technique where hacker executes malicious SQL queries on the database server through a web application to either gain access over the sensitive information or on the database [1].

2.1. SQL injection: Threats

1. Identity Spoofing: It refers to an attack in which the identity of some other person is used to accomplish the task or a goal.
2. Changing the data value: It is an attack in which the hacker changes the value of the data

stored in the database. For example, a hacker can enter an online shopping site and then changes the price of the products.

3. Modifying the database: In this attack an attacker either deletes the data or it completely replaces a existing data with a new data value.
4. Gaining access over administrative privileges: After gaining successful access to the database and to the database network, an attacker tries to gain access over the administrative privileges also.
5. Denial of Service: when the system or server is overloaded with the multiple bogus requests which cannot be handled by the system which results in temporarily halt in the service and thus user is unable to access the system or server.
6. Gains access over the highly sensitive information: Once the hacker gains access on the network, the attacker gains access to all the sensitive information stored in the database.
7. Destroys the existing data stored in the database: Once the attacker is successful in gaining access to the database then it can easily destroy all the sensitive information stored in that database.
8. Attack machine‘s performance: An attacker halts all the important transactions performed by the system.

2.2. SQL injection: Attacks

The main attacks associated with the SQL injection are:

1. Authentication Bypass: In this the attacker without providing valid username and password could successfully gain access over the network by manipulating the SQL commands.[2]
2. Leaking sensitive information: After gaining unauthorized access to the database, the hacker gets illegal access to all the sensitive information stored in the database. [1]
3. Loss of data integrity: In this attack the attacker not only modifies the content of the database but also add malicious content to the database. [1]
4. Loss of availability of data: once the attacker gets complete access to the system or the server, he/she can delete the important data from the database which results in huge loss of data and leads to data unavailability.[2]

2.2. SQL injection: Types

1. Tautology attack: Tautology attack is used to inject malicious code to the SQL statements which will always evaluates to true. This technique is most commonly used to bypass the authentication part.
   For example: SELECT * FROM user WHERE Username=‘‘ or 1=1—AND pass1=‘‘ [4]
2. Illegal /logically incorrect queries: In this type to attacks the database sends error message to the user in case of entering wrong information in SQL statements. This error message is used by the hacker to attack the database as the error message from the database gives some information about the database. The information given in the error message could be the database name or the entities of the database, attributes of the database, etc.
   For example: the error message for sending a wrong password may be like:
   SELECT * FROM user WHERE username=‖‖ AND password=‖123‖ or 1=1; [4]
3. Using piggy-backed queries: Previously mentioned attack types try to gain unauthorized access to the applications database. If an additional queries are piggy-backed in the input statements using special characters such as — —, ‖—— or —;‖ hackers can modify or delete the database.
   For example: SELECT * FROM user WHERE id=123; drop table user; [3]
4. Blind injection: Programmers try to hide database information from the attacker in case of illegal or incorrect queries. Blind SQL injection can be used to obtain access over sensitive information present in the database by asking series of true of false statements through SQL Queries. [3]

III. SQL INJECTION QUERIES

3.1. SQL injection Query

A user first enters the username and password then a dynamic SQL query is generated based on the user input in the SQL statements. A user is first authenticated and then directed to the user requested page.

```
Malicious SQL Query:
SELECT user FROM users WHERE username='abc'
OR 1=1- -'AND password-" abc""
```

3.2. Query for updating table

Through the following malicious SQL query attacker modifies the content of the table:

```
Malicious SQL Query:
"abc';UPDATE customer SET
email='abc@xyz.com'WHERE
email='john@xyz.com';- -""
```

3.3. Query for Adding new records

Through the following malicious SQL query attacker can add new record to the database:

```
Malicious SQL Query:
"abc'; INSERT INTO
customers=('email','password','login_id','last_name
')VALUES('john@xyz.com','abc','john','haden');--
```

3.4. Query for identifying table name
Through the following malicious SQL query attacker can identify the table name:

Malicious SQL query:
"abc' AND 1=(SELECT * FROM mytable);--"

3.5. Query for deleting the table
Through the following malicious SQL query an attacker can delete the table:

Malicious SQL query:
"abc'DROP TABLE creditcard;--"

IV. SQL INJECTION TOOLS

Following tools are used for exploiting the vulnerabilities of SQL injection:

4.1 BSQL(Blind SQL)Hacker: It supports blind SQL attacks, time based SQL injection, error based SQL injection and deep SQL injection attacks. It allows attacker to exploit queries on the database.it is fast and also supports multithreading and GUI.

4.2 Marathon Tool: It is used for performing time based SQL injection attack. It has flexible configuration in injection and adds variable and values in cookies.

4.3 SQL Power injector: This tool is used to exploit SQL vulnerabilities on a web page. It supports blind SQL injection and supports SQL server, Oracle, MYSQL, Sybase server and DB2 but when used with normal database it supports any existing database.

4.4 Havji: It is an automated SQL injection tool which allows an attacker to perform illegal SQL queries on a web page by performing backend fingerprinting, retrieving users and password hashes and by executing commands on the operating system.

V. PREVENTING SQL INJECTION ATTACKS

5.1 Minimizing the privileges: Security aspects must be prioritized and adequate steps must be taken in the initial stage of the development cycle. So that the developer becomes familiar with all the security aspects and this will help them to develop a much secured product.

5.2 Implementation of consistent coding standards: Well-planned security infrastructures and set of standards and policies must be laid down.[1]

5.3 Using encryption technique: All the data stored in the database and all the attributes of the table must be encrypted for the security reasons.[]

5.4 Using Client side validation: Using Client side validation such as JavaScript, limiting the input size, limiting the input character type, use of special characters ,etc. can be used for preventing SQL injection attacks in web based applications.

CONCLUSION

SQL injection is a web based attack that allows an hacker to obtain an illegal access to the database and to all other sensitive information by manipulating SQL injection commands. These SQL commands can be used to either modify to or to delete the data stored in the database. In this paper we broadly discussed SQL injection its threats, its attacks, queries and prevention strategies.

ACKNOWLEDGMENTS

I would like to take this opportunity to express my profound gratitude and deep regard to my guide Mrs. Smita Singh, for her exemplary guidance, valuable feedback and constant encouragement throughout the duration of the project. Her valuable suggestions were of immense help throughout my project work.This research was supportedbyMody University of Science and Technology. I would also like to thank my friends and family who provided insight and expertise that greatly assisted the research,.

REFERENCES

[1] Nanhay Singh, MohitDayal and R.S. Raw, Suresh Kumar, ―SQL Injection: Types, Methodology, Attack Queries and Prevention‖, International Conference on Computing for Sustainable Global Development, ,pp.2872–2876,2016.

[2] P. Saravana Kumar, M.Parvathi and M.kanmani, ― Efficient Method for Preventing SQL Injection Attacks on Web Applications Using Encryption and Tokenization‖, Study on SQL injection-threats, attacks, types, prevention techniques and tools International Journal of Latest Trends in Engineering and Technology, vol.4, pp. 75-84, 4 November 2014.

[3] Srinivas Avireddy, VaralakshmiPerumal, Narayan Gowraj, Ram Srivatsa Kanna, PrashanthThinakaran, SundaravadanamGanapathi, Jashwant Raj Gunasekaran and SruthiPrabhu ― Random4: An Application Specific Randomized Encryption Algorithm to prevent SQL injection‖, International Conference on Trust, Security and Privacy in Computing and Communications,vol.26,pp.1327-1333,10 November 2012

[4] Monali R. Borade, Neeta A. Deshpande, ―Web Serviced Based SQL Injection Detection and Prevention System for Web Applications‖, vol.4, pp.418-426, October 2014.

[5] Dr. Amit Chaturvedi, ShailendraBagdi, VikasChoudhary,―Analysis of SQL Injection Attacks and Vulnerabilities‖,International Journal of Advanced Research in Computer Science and Software engineering,vol.6,pp. 106-110, 3 March 2016.

[6] R.P. Mahapatra, Subhi Khan, ―A Survey on SQL injection Countermeasures‖,International Journal of computer Science and engineering survey, vol.3,pp.55-74, June 2012.

[7] LwinKhinShar, heeBengKaun Tan, ― Defeating SQL injection,‖,IEEE computer society,vol. 46,pp.69-77, March 2013.