International Journal of Engineering and Techniques - Volume 4 Issue 2, Mar – Apr 2018

RESEARCH ARTICLE OPEN ACCESS

# SQL Query Injection a Hazard Using Web Application

Divya.M [1] Monisha.S[2] Reena .R[3], Kapila vani.R.K.[4]

Department of Computer Science and Engineering, Prince Shri Venkateshwara Padmavathy, Engineering College, Chennai, India[1,2]

Assistant Professor, Department of Computer Science and Engineering, Prince Shri Venkateshwara Padmavathy Engineering College, Chennai, India.[3]

Assistant Professor, Department of Computer Science and Engineering, Prince DR.K Vasudevan College of Engineering and Technology, Chennai, India.[4]

**Abstract:**

Most information systems and business applications built nowadays have a web frontend and they need to be universally available to clients, employees and partners around the world, as the digital economy is becoming more and more prevalent in the global economy. These web applications, which can be accessed from anywhere, become so widely exposed that any existing security vulnerability will most probably be uncovered and exploited by hackers. SQLi and XSS allow attackers to access unauthorized data (read, insert, change or delete), gain access to privileged database accounts. The data may contain credit card numbers, account numbers, social security numbers, user names, passwords, email accounts, etc. These goods have a huge demand in the underground economy, which indicate that they have a higher cost/benefit ratio compared to other types of attacks.

*Keywords* — **Database, SQL Injection, DBMS assert , Bind Variables , Information Disclosure , Authentication bypass.**

## I.INTRODUCTION

Structured Query Language (SQL) is a language that is used to query, operate, and administer database systems such as Microsoft SQL Server, Oracle, or MySQL. The general use of SQL is consistent across all database systems that support it. Web Intrusion refers to any attempt to threaten the confidentiality or availability of data in web application. SQL injection is a code injection technique used to attack data-driven applications, in which malicious SQL statements are inserted into an entry field for execution (e.g. to dump the database contents to the attacker). It consists of insertion or injection of a SQL query via the input data from the client to the application. A successful SQL injection statement can read sensitive data from the database, modify database data (such as Insert/Update/Delete), execute administration operations on the database (such as shutdown the DBMS), recover the content of a given file present on the DBMS file system.The data may contain credit card numbers, account numbers, social security numbers, user names, passwords, email accounts, etc. These goods have a huge demand in the underground economy, which indicate that they have a higher cost/benefit ratio compared to other types of attacks.SQL injection can be used to perform the following types of attacks:

- Authentication Bypass: This attack allows an attacker to log on to an application, potentially with administrative privileges, without supplying a valid username and password.
- Information Disclosure: This attack allows an attacker to obtain, either directly or indirectly, sensitive information in a database

An Oracle database is a collection of data treated as a unit. The purpose of a database is to store and retrieve related information. A database server is the key to solving the problems of information management. In general, a server reliably manages a large amount of data in a multiuser environment so that many users can concurrently access the same data. All this is accomplished while delivering high performance. A database server also prevents unauthorized access and provides efficient solutions for failure recovery.Most web applications have critical bugs (faults) affecting their security, which makes them vulnerable to attacks by hackers and organized crime. To prevent these security problems from occurring it is of utmost importance to understand the typical software faults. This paper contributes to this body of knowledge by presenting a field study on two of the most widely spread and critical web application vulnerabilities: SQL Injection and XSS.

. There is no need for peak workloads, because capacity can be easily added or reallocated from the resource pools as needed. The database has logical structures and physical structures. Because the physical and logical structures are separate.The SQL Injection is the application layer technique in which the hacker can steal the data from the organisation.The SQL Injection is the hacking technique in which the hacker can pass the SQL commands through web application for execution

at backend database.

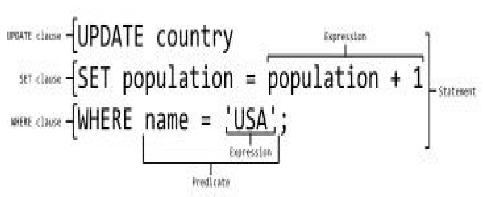
## II.BACKGROUND AND RELATED WORK

An Oracle database system identified by an alphanumeric system identifier or SID comprises at least one instance of the application, along with data storage. An instance identified persistently by an instantiation number comprises a set of operating-system processes and memory structures that interact with the storage. (Typical processes include PMON (the process monitor) and SMON (the system monitor).) Oracle documentation can refer to an active database instance as a "shared memory realm".

Users of Oracle databases refer to the server-side memory-structure as the SGA (System Global Area). The SGA typically holds cache information such as data-buffers, SQL commands, and user information. In addition to storage, the database consists of online redo logs (or logs), which hold transactional history. Processes can in turn archive the online redo logs into archive logs (offline redo logs), which provide the basis (if necessary) for data recovery and for the physical-standby forms of data replication using Oracle Data Guard.

If the Oracle database administrator has implemented Oracle RAC (Real Application Clusters), then multiple instances, usually on different servers, attach to a central storage array. This scenario offers advantages such as better performance, scalability and redundancy. However, support becomes more complex, and many sites do not use RAC. In version 10*g*, grid computing introduced shared resources where an instance can use (for example) CPU resources from another node (computer) in the grid. The Oracle DBMS can store and execute stored procedures and functions within itself. PL/SQL (Oracle Corporation's proprietary procedural extension to SQL), or the object-oriented language Java can invoke such code objects and/or provide the programming structures for writing them.

SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standards (ISO) in 1987.[12] Since then, the standard has been enhanced several times with added features. Despite these standards, code is not completely portable among different database systems, which can lead to vendor lock-in. The different makers do not perfectly adhere to the standard, for instance by adding extensions, and the standard itself is sometimes ambiguous.The language elements:



```
UPDATE country
SET population = population + 1
WHERE name = 'USA';
```

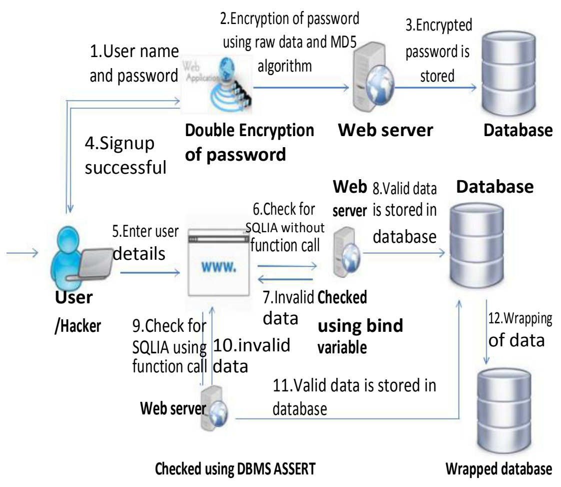
## III.CURRENT METHODOLOGY

Existing system emphasizes on several security aspects, including access privacy. As network security practitioners put more resources and effort in to defending against SQL INJECTION ATTACKS, hackers will develop and deploy the next generation of SQLIA (SQL Injection Attacks) bonnets with different control architecture. A SQL injection attack is an attack that is aimed at subverting the original intent of the application by submitting attacker-supplied SQL statements directly to the backend database. Through this a hacker can easily enter into a user account and access their own information. Thus, he can easily execute oracle function from the select statement. If a DBA knows the user password, he can easily access user account without user permission. Traditionally, as soon as confidentiality becomes a concern, data is encrypted before outsourcing to a service provider.In the existing system, the faults in the web page are identified using join algorithm but it is not prevented. The authentication process is not secured since there is only single level encryption. User can add SQL injection attacks to the database. It allows execution of oracle function or custom function using the select statement.The advantage of this model is:the sql injection attack is identified,single level security is provided.The disadvantage of this module is: Bypasses the login authentication,Selects secured information from database tables,Security mechanisms are not that efficient in existing system,Using function call all the field data can be viewed by the hackers,Only single level security for web databases have been made possible.

## IV. PROPOSED SYSTEM

The main objective of the proposed system is to keep database in a secured manner under serious SQL injection attacks and to analyse the principle of SQL attacks. It provides safety methods to both users as well as administratorsThe module for the proposed system is given below:

1.encryption process using raw data and MD5 algorithm.
2. SQLIA without function calls using bind variable technique.
3.SQLIA with function calls using DBMS assert algorithm.
4.Wrapped using wrapping function.


1.User name and password
2.Encryption of password using raw data and MD5 algorithm
3.Encrypted password is stored
Double Encryption of password
Web server
Database
4.Signup successful
Web
server
8.Valid data is stored in database
Database
6.Check for SQLIA without function call
5.Enter user details
www.
User /Hacker
7.Invalid data
Checked using bind variable
9.Check for SQLIA using function call
10.invalid data
12.Wrapping of data
11.Valid data is stored in database
Web server
Checked using DBMS ASSERT
Wrapped database

The system architecture is the conceptual design that defines the structure and behaviour of a system. An architecture description is a formal description of a system organized in a way that supports reasoning about the structural properties of the system. It is the first part of system design. The main aim of the input design is covering user oriented descriptions of the input to the computer oriented form. All the inputs are converted into a computer based format. The goal of designing input data is to make data entry easier and free errors as possible.

**Encryption Process Using Raw Data And MD5 Algorithm:** The Goals of this modules is to encrypted the password using raw data algorithm and MD5 algorithm. Here when the user enter the input password, the password is encrypted using raw data algorithm. Then the password is converted into hexadecimal bit using MD5 algorithm.The converted information is stored onto the database in encrypted form.so, the hacker cannot view the user details in its original form. The hacker cannot enter the user login by using tricky queries.The output of this module is encrypted data bits stored in database.

**SQLIA without function calls using bind variable technique**: The Goal of this modules is to check SQL Injection attack in the login page. Here the user details which are supposed to be the input for this module are checked for SQLIA (SQL Injection Attack) using bind variable. If it is not a SQL injection code the process proceeds. If it is a SQL injection code the attacks are avoided.The main idea of this module is the identification of SQLIA.So the bypass authentication will be avoided by using bind variable technique.

**SQLIA with function calls using DBMS assert algorithm**: The Goal of this module is to validate and ignore the fake action that the attacker try to execute. The attackers try to insert a number of additional illegal SQL statements into the normal structure of the dynamic SQL statement. This might result in the execution of normal SQL statements that client sends along with these additional SQL statements which attackers construct. These additional SQL statements are often key operations to the database such as deleting the data table, modifying table fields, adding data and deleting data. In this type of SQLIA attackers inject a statement of the form ―UNION < injected query >‖.

By using function call statements attackers can retrieve the table name and information from a specified table.To overcome this problem the DBMS_ASSERT algorithm is proposed.Using this algorithm we avoid the Function call injection problems.The product details are checked for function call injection and SQLIA.They are valiated using the DBMS_ASSERT algorithm.If there are no intrusions the details are stored in the database.If there is any intrusion then it is reported to the webpage.It validates and ignores the fake actions that the attackers try to execute.

**Wrapped using wrapping function:** The Goal of this module is to hide the database scheme information and the source code from the hackers using wrapping function. The database schema information and the source codes are to be secured and hidden from the hackersThis is done using the process of wrapping.It hides all the information from hackers by using the wrapped algorithm in a small package.The wrapped information is then processed to be stored in the database.Because of this module, the hacker can not hack the data using the URL.

**Algorithm used:**
The algorithm mainly used in this project is raw data algorithm,MD5 algorithm and DBMS assert.The first algorithm is used to encrypt the password in raw data.

```
ALGORITHM 1
Input: Password`
Output: Encrypted Password in raw data form.
BEGIN
Step 1: Convert the string into raw data using the function STRING_TO_RAW().
Step 2: Use the default key A1B2C3D4E5F6G7H8.
Step 3: Padding and chaining are handled using encryption mode.
Step 4: The encryption process is done using the following function:

DBMS_CRYPTO.ENCRYPT(UTL_118N.STRING_TO_RAW
(data,'AL32UTF8'),encryption
mode,UTL_118N.STRING_TO_RAW(KEY,'AL32UTF8``));
END
```

Then, after encryption of password into raw data.we double encrypted the password and stored it in the database.

```
ALGORITHM 2
Input: Encrypted password in raw data form.
Output: Double encrypted
password BEGIN
Step 1: Append Padding Bits.The original message is —padded‖(extended)so that its length(in bits) is congruent to 448,modulo 512.
    Step 1.1: The orginal message is always padded with one bit —1‖ first.
    Step 1.2: Then zero or more bits —0‖ are padded to bring the length of the message up to 64 bits fewer than a multiple of 512.
Step 2: Append Length.64 bits are appended to the end of the padded message to indicate the length of the original message in bytes.
    Step 2.1: The length of the original message in bytes is converted to its binary format of 64 bits. If overflow happens, only the low-order 64 bits are used.
    Step 2.2: Break the 64-bit length length into 2 words(32 bits each).
    Step 2.3: The low-order word is appended first and followed by the high-order word.
Step 3:Initialize MD Buffer.MD5 algorithm requires a 128-bit buffer with a specific initial value.
    Step 3.1: The buffer is divided into 4 words(32 bits each),named as A,B,C and D.
        Word A is initialized to: 0x67452301
        Word B is initialized to: 0xEFCDAB89.
        Word c is initialized to: 0x98BADCFE.
        Word d is initialized to: 0x10325476.
Step 4: Processing Message in 512-bit Blocks. This is the main step of MD5 algorithm, which loops through the padded and appended message in blocks of 5122 bits each. For each input blocks, 4 rounds of operations are performed with 16 operations in each round.
Step 5: Output. The contents in buffer words A,B,C,D are returned in sequence with low-order byte first.
```

DBMS assert algorithm will be used when the user enter the product details and check whether the product is valid. The product details are checked for function call injection and SQLIA. They are validated using the DBMS_ASSERT algorithm.

```
ALGORITHM 3
Input: Product details.
Output: Validated product details.
BEGIN
Step 1:The queries are passed through the DBMS ASSERT package.
Step 2:Each query is processed word by word and the keywords are checked for intrusions.
Step 3:An error is reported to the web application if there is an intrusion.
Step 4: If there is no intrusion then the process is continued.
END
```

## VI.CONCLUSION AND FUTURE WORK

In this paper, we analyse the query attacks and web intrusions in the web database and find the prevention measures. The faults in the web page is identified and validated using various techniques and algorithms.Hackers cannot bypass login authentication since we provide double encryption.Hacking thesource code from the server are prevented using wrapping function.Thus the proposed system will prevent told he intrusions and query attacks in web page.

The proposed system —Prevention of web intrusions and query attacks on web databse@ is implemented in oracle 10 g. This can be further enchanced by reducing the disk space required for storing the data .The information should be wrapped in a more secured manner so that no one can access it. The database protection can be enhanced in the future.

## REFERENCE

[1] P. Anbalagan and M. Vouk, —Towards a Unifying Approach in Understanding Security Problems,‖ Proc. Int'l Symp. Software Reliability Eng., pp. 136-145, 2009.

[2] R. Chillarege, I.S. Bhandari, J.K. Chaar, M.J. Halliday, D. Moebus, B. Ray, and M. Wong, —Orthogonal Defect Classification—A Concept for In-Process Measurement,‖ IEEE Trans. Software Eng., vol. 18, no. 11, pp. 943-956, Nov. 1992.
[3] A. Avizienis, J.C. Laprie, B. Randell, and C. Landwehr, —Basic Concepts and Taxonomy of Dependable and Secure Computing,‖ IEEE Trans. Dependable and Secure Computing, vol. 1, no. 1, pp. 11- 33, Jan.-Mar. 2004.
[4] Acunetix Ltd., —Is Your Website Hackable? Do a Web Security Audit with Acunetix Web Vulnerability Scanner,‖ http://www.acunetix.com/security-audit/index/, May 2013.
[5] G. Alvarez and S. Petrovic, —A New Taxonomy of Web Attacks Suitable for Efficient Encoding,‖ Computers and Security, vol. 22, no. 5, pp. 435-449, July 2003.
[6] P. Anbalagan and M. Vouk, —Towards a Unifying Approach in Understanding Security Problems,‖ Proc. Int'l Symp. Software Reliability Eng., pp. 136-145, 2009.
[7] J. Dura˜es and H. Madeira, —Emulation of Software Faults: A Field Data Study and a Practical Approach,‖ Trans. Software Eng., vol. 32, pp. 849-867, 2006.
[8] A. Adelsbach, D. Alessandri, C. Cachin, S. Creese, Y. Deswarte, K. Kursawe, J.C. Laprie, D. Powell, B. Randell, J.Riordan, P. Ryan,W. Simmonds, R. Stroud, P. Verissimo, M. Waidner, and A.Wespi, —Conceptual Model and Architecture of MAFTIA,‖ Project IST-1999-11583
[9] JM. Cukier, R. Berthier, S. Panjwani, and S. Tan, —A Statistical Analysis of Attack Data to Separate Attacks,‖ Proc. Int'l Conf. Dependable Systems and Networks, pp. 383- 392, 2006.
[10] J. Christmansson and R. Chillarege, —Generation of an Error Set That Emulates Software Faults,‖ Proc. IEEE Fault Tolerant Computing Symp., pp. 304-313, 1996.
[11] E. Bertino, Data Protection from Insider Threats. San Rafael, CA, USA: Morgan Claypool, 2012.
[12] A. Seshadri, A. Perrig, L. van Doorn, and P. Khosla, —SWATT:SoftWare-based attestation for embedded devices,‖ in Proc. IEEE Symp. Secur. Privacy, May 2004, pp. 272–282.
[13] C. Cowan, P. Wagle, C. Pu, S. Beattie, and J. Walpole, —Buffer overflows: Attacks and defenses for the vulnerability of the deca- de,‖ in Proc. DARPA Inf. Survivability Conf. Exposition, 2000, vol. 2, pp. 119–129.
[14] C. Cadar and K. Sen, —Symbolic execution for software testing:Three decades later,Commun. ACM, vol. 56, no. 2, pp. 82–90, Feb.2013.
[Online]. 4th Int. Conf. Softw. Eng., May 2007, pp. [illegible] Majumdar and K. Sen, —Hybrid concolic testing,‖
[15] W. G. Halfond, J. Viegas, and A. Orso, —A classification of SQL injection attacks and countermeasures,‖ in Proc. IEEE Int. Symp. Secure Softw. Eng., 2006, vol. 1, pp. 13–15.

## BIOGRAPHY

**Monisha S** is a student doing B.E degree in computer science and engineering in Prince shri venkateshwara padmavathy Engineering college, Chennai. Her research interest includes Cryptography, network security, cyber security,database security.

**Divya M** is a student doing B.E degree in computer science and engineering in Prince shri venkateshwara padmavathy Engineering college, Chennai. Her research interest includes mobile computing, network security, cyber security.

**Reena R** is an assistant professor in computer science department at Prince Shri Venkateshwara padmavathy Engineering college, Chennai. She is a M.E graduate. Her research interest includes computer graphics, internet programming, mobile computing.

**Kapila Vani. R. K.** as Assistant Professor in computer science department at Prince Dr.K vasudevan College of Engineering and Technology,Chennai. She is a M.E graduate. Her research interest includes compiler design, Theory of computation and Software project management.