A Survey on SQL Injection Attacks and Prevention Methods

Ashwath Keshav Hegde[1], P N Jayanthi[2]

[1]*Student, Dept. of ECE, R V College of Engineering, Bangalore, India*
[2]*Professor, Dept. of ECE, R V College of Engineering, Bangalore, India*

---***---

Abstract – *Number of devices connected to internet are increasing day by day. Number of users for web applications is also increased rapidly. Most of the organization will have their website to give information to the users or to provide the service. Database is necessary to store data related to users or to store any information which users are served. SQL is used widely to communicate with the database. In SQL injection attack, malicious SQL statement is executed on the database by the attacker. SQL injection is very serious security threat as it can be employed to steal the content of database, change the values stored in the database, even whole database can be erased. In most of organizations content of database are very confidential and have financial importance for the organization. This review shows how the attack can be mitigated effectively.*

Key Words: SQL, Data, Database, Internet, SQL query, Server

1.INTRODUCTION

A database is an organized collection of data. SQL stands for Sequential query language. SQL is used for executing commands on the database, it is used for data definition, data manipulation and data control. SQL Injection attacks are used widely, because of the fact that most of the information is stored in the database. SQL is used widely to perform operations on the database. So, it is very effective attack to harvest confidential information from an organization [1]. Server takes the request from user and gives back relevant response. Content of request sent by user is used by the server to create the response. If server uses the inputs without verifying it, to query a statement from database, then there will be chance of SQL injection [2]. It can be used to steal all the data in the database, to change values in the database, or even to erase the database [3]. Which can corrupt the database or complete loss of database. Different methods to prevent SQL injection are presented in this paper. SQL statement which takes the input from other source or which uses data obtained from external source are major contributors of the cause. Prevention of SQL injection attack is crucial in any web application [4]. There can be various intentions [5] behind the attack, it is explained in the paper. SQL injection caused huge loss to many organizations till date, so it is considered as one of the most used cyber-attack [6]. Even simple logical error can lead to various attacks on the system and attacker can get into the system [7]. Due to advancement in machine learning techniques, it is used to detect SQL injection attacks effectively [8].

1.1 Intention for Attack

SQL injection attacks can be classified based on intention or goal of attack. There can be various motivations and intention for an attack [5], below are few important intentions for the attack.

- Identifying injectable parameters: The parameters used by the application are analyzed. The possible parameters that are vulnerable to SQL injection attack identified and which helps for possible attack.
- Database finger-printing: Type and version of database are found out by the attacker. Using these for database specific attacks.
- Finding database schema: Attacker often needs to know database schema information to extract data properly from a database, such as names of table and its column, column data types and relation between the tables.
- Extracting data: The values stored in the database are extracted by the attacker. This data can be very sensitive. This is the major intent of SQL injection attacks by majority of attackers.
- Adding or modifying or deleting data: Changing the values in database purposefully or randomly will corrupt the database. This will make the application to behave abnormally.
- Bypassing authentication: goal of this type of attack is to get rights and privilege to the application by unauthorized person.
- Executing remote commands: Executing some arbitrary or some particular command on the database. Which can even be used to delete content of database, or to corrupt complete database.

1.2 Detection and Prevention Techniques

- Tautology Checker
  Static analysis is used to prevent tautology attack [9]. Arithmetic and logical loops are used to check for possible SQL injection. Abstract model of a source program is used in the framework, that takes inputs from user and constructs SQL queries. Particularly, the set of SQL queries which

a program could generate as a finite state automaton is approximated. The framework then applies some novel checking algorithms on this automaton. Which indicate or verify lack of security violations in the program of application. This method is not suitable for finding out other SQL injection attacks.

- Predictive Errors by the Feature of the Single Character

  Probabilistic model to the SQL Injection attack is designed [10]. Which tries to minimize the predictive error in the SQL Injection attack detection. Twenty single characters are taken as candidates. A function similar to a sigmoid function is designed to evaluate the features.
- Validation using Parse Tree

  A parse tree is a widely used data structure, it is used to represent a statement in the parsed representation [11]. Grammar of the statement's language is needed to Parse a statement. Parsing two statements then using the parsed tree of both the statements for comparison, it can be determined whether the two queries are equal. This method is easy to integrate with even existing software. This implementation reduces the effort of the programmer, as it captures both the intended query and actual query with least changes required by the coder.
- AMNESI

  AMNESIA is a technique that detects and prevents SQL injection attacks. It combines static analysis and runtime monitoring. This method is very effective and efficient against SQL injection attacks. Initially hotspots are identified in the application code, then SQL query model is built [12]. For each identified hotspot to runtime monitor call is added. Dynamically generated queries are checked against the SQL query model at the runtime. Queries that violate the model are rejected in the process.
- SQL Check

  The approach is validated with SQLCHECK, it is an implementation for the setting of SQL command injection attacks. SQLCHECK is evaluated on real-world web applications with real-world attack data as input which are systematically compiled [13]. SQLCHECK produced least false positives and false negatives so F1 score is very high. Runtime overhead is very low and it can be applied straightforwardly to web applications written using different programming languages.
- Preventing SQL Injection Attacks using SQLrand

  Practical protection method to prevent SQL injection attacks is presented [14]. The concept of randomizing instruction-set to SQL is applied, to generate instances of the language which cannot be predicted by the attacker. The queries injected by the attacker to the application will be detected and terminated by the database parser. This technique imposes almost negligible performance overhead in query processing and it can be easily integrated with existing systems that are using databases.
- Preventing SQL Injection Attacks with Stored Procedures

  This method [15] eliminates the occurrence of SQL injection attacks by combining static application code analysis with runtime validation. Stored procedure parser is designed in the static part, this parser is used to instrument the necessary statements for any SQL statement that uses user inputs. Main purpose of this is to compare the original SQL statement structure with the statement that includes user inputs. The deployment of this technique can be automated and can be only when necessary.
- Combinatorial Approach

  The approach [16] against SQL injection attacks is based on Signature based approach, that is based on validation of input to solve security vulnerability. Three modules are designed to detect security issues.
  - Monitoring module

    Every SQL query is being checked before the execution, to prevent possible attack. This unit will decide whether to send SQL statement to the database for execution.
  - Analysis module

    Hirschberg algorithm is used to compare SQL statement with predefined keywords.
  - Auditing module

    If there are any suspicious statements found then it stops the transaction and audits the report of attack.

2. CONCLUSIONS

SQL injection is very widely used attack and it is very powerful. Detecting and preventing possible attack is very important aspect of security of any application. As SQL injection attacks have caused huge business or economic impact on many organizations, so it very important to prevent any such attacks. This paper presents a wide survey of the SQL Injection attacks. The ways which SQL injection attacks can be performed are presented. The attacks are categorized based on the way in which it is performed based on the vulnerability. Different types of SQL Injection detection and prevention techniques are also discussed. Advantages and disadvantages of the same are discussed.

REFERENCES

[1] J. Fonseca, M. Vieira and H. Madeira, "Vulnerability & attack injection for web applications", IEEE/IFIP International Conference on Dependable Systems & Networks, Lisbon, 2009, pp. 93-102, 2009

[2] R. A. Katole, S. S. Sherekar and V. M. Thakare, "Detection of SQL injection attacks by removing the parameter values of SQL query", 2nd International Conference on Inventive Systems and Control (ICISC), Coimbatore, 2018, pp. 736-741, 2018

[3] Christey S, "Unforgivable vulnerabilities", Black Hat Briefings, 2007

[4] Mohammad Abu Kausar, Mohammad Nasar, Aiman Said, "SQL Injection Detection and Prevention Techniques in ASP.NET Web Application", International Journal of Recent Technology and Engineering, 2019

[5] W. G Halfond, T. Viegas, and A. Orso, "A Classification of SQL injection Attacks and Countermeasures", In Proc. of the Intl. Symposium on Secure Software Engineering, Mar. 2006

[6] H. Gupta, S. Mondal, S. Ray, B. Giri, R. Majumdar and V. P. Mishra, "Impact of SQL Injection in Database Security", 2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE), Dubai, United Arab Emirates, pp. 296-299, 2019

[7] R. M. Pandurang and D. C. Karia, "Impact analysis of preventing cross site scripting and SQL injection attacks on web application", IEEE Bombay Section Symposium (IBSS), Mumbai, 2015, pp. 1-5, 2015

[8] S. Abaimov and G. Bianchi, "CODDLE: Code-Injection Detection with Deep Learning", in IEEE Access, vol. 7, pp. 128617-128627, 2019

[9] G Wassermann, Z. Suo, "An Analysis Framework for Security in Web Applications. In Proceedings of the FSE Workshop on Specification and Verification of Component Based Systems (SAVCBS 2004)", pp 70-78, 2004

[10] T. Matsuda, D. Koizumi, M. Sonoda and S. Hirasawa, "On predictive errors of SQL injection attack detection by the feature of the single character", IEEE International Conference on Systems, Man, and Cybernetics, Anchorage, AK, pp. 1722-1727, 2011

[11] Buehrer, G. T, Weide. B. W, Sivilotti, P. A. G, "Using parse tree validation to prevent SQL injection attacks", Proceedings of the 5th International Workshop on Software Engineering, 2005

[12] William G. J. Halfond, Alessandro Orso, "Preventing SQL injection attacks using AMNESIA", In Proceedings of the 28th international conference on Software engineering (ICSE '06). Association for Computing Machinery, New York, NY, USA, 795–798, 2006

[13] Zhendong Su, Gary Wassermann, "The essence of command injection attacks in web applications", SIGPLAN Not. 41, 372–382, 2006

[14] S. W. Boyd, A. D. Keromytis, "SQLrand: Preventing SQL Injection Attacks", In Proceedings of the 2nd ApplIed Cryptography and Network Security Conference, pages 292- 302, 2004

[15] 15. K. Wei, M. Muthuprasanna, Suraj Kothari, "Preventing SQL injection attacks in stored procedures", Australian Software Engineering Conference (ASWEC'06), Sydney, NSW, pp. 8 pp.-198, 2006

[16] 16. R. Ezumalai, G. Aghila, "Combinatorial Approach for Preventing SQL Injection Attacks", IEEE International Advance Computing Conference, Patiala, pp. 1212-1217, 2009