EVALUATING THE SECURITY RISKS AND PROTECTION STRATEGIES FOR SQL INSERT QUERIES AGAINST INJECTION ATTACKS

Inomjon Yarashov
Mirabbos Akbarov
Diplomat university
The University of World Economy and Diplomacy
e-mail: iyarashov@uwed.uz
https://doi.org/10.5281/zenodo.15396515

Abstract. The study explores the security risks associated with the use of Insert queries generated from user-provided data. In order to evaluate the vulnerability of Insert queries to SQL injection attacks and identify effective defense strategies, we conducted an experiment. This experiment demonstrated how an attacker could extract sensitive information, such as the database user credentials, details about the database itself, and the version of the server being used. Moreover, during a root user session, access to the attacked server's file system was achievable. Our findings emphasize that, to safeguard against such vulnerabilities, it is crucial to implement stringent filtering of user input data.

Keywords: SQL injection, Insert queries, user input validation, security risks, database vulnerabilities, protection strategies, server access, data filtering.

Introduction

SQL injection attacks have emerged as one of the most widely exploited vulnerabilities in web applications, particularly targeting databases. These attacks are based on manipulating SQL queries to execute arbitrary commands in a database, often compromising sensitive data and leading to unauthorized access. Various algorithms and methodologies have been proposed to detect, prevent, and mitigate the effects of SQL injections, with many focusing on common SQL query structures such as SELECT, UPDATE, and DELETE queries [1-9]. However, there has been little research addressing the potential vulnerabilities associated with INSERT queries in the context of SQL injection attacks.

INSERT queries, which are used to add new records to a database, are often overlooked in security discussions, leading to a gap in understanding the risks they present when improperly validated user input is involved. Despite their critical role in data insertion, the possibility of exploiting INSERT queries through SQL injection remains largely unexplored. This study aims to bridge this gap by investigating the security implications of SQL injections in INSERT queries.

To assess the potential damage caused by SQL injection in these queries, we propose using a method based on measuring delays in SQL query execution. This approach will help identify patterns and anomalies that indicate malicious SQL injections. Additionally, we will explore various countermeasures to mitigate the risks posed by SQL injections in INSERT queries, focusing on techniques such as input validation, parameterized queries, and other best practices for secure database management.

Through this research, we seek to contribute to the broader field of SQL injection defense by highlighting vulnerabilities in INSERT queries and offering practical solutions to enhance database security. The findings of this study are expected to provide valuable insights for developers and security professionals in designing more robust systems and defending against these pervasive attacks.

Theoretical analysis

The works [10-16] provide extensive descriptions of SQL injection examples, primarily focusing on scenarios that involve the use of the UNION clause to combine multiple queries to the database. These attacks assume that data obtained from the database query can be outputted, allowing attackers to influence the results of the original SQL query [17-26]. However, less commonly discussed are algorithms describing character-by-character SQL injection attacks in SELECT, UPDATE, and DELETE queries, which use latency measurements as a technique for identifying vulnerabilities [27-36].

This form of SQL injection is particularly relevant when the attacker cannot directly manipulate the data returned by queries, when no error messages are generated, and when the UNION clause cannot be employed. In such cases, the attacker also lacks detailed knowledge of the database structure. Despite these limitations, a successful attack remains feasible if the attacker can insert specific restrictions into the query, narrowing down the results to a single record. Typically, these SQL injection attacks are applicable to SELECT, UPDATE, and DELETE queries, where conditions are added after the WHERE clause.

To effectively counter these types of attacks, it is essential to implement proper filtering of the data involved in the generation of database requests. Filtering ensures that only valid and safe input is included, preventing attackers from exploiting query vulnerabilities.

INSERT queries, however, present a unique challenge because they do not contain WHERE clauses, making data retrieval from the database impossible unless the table contains only a single record. This limitation reduces the risk of data extraction through INSERT queries. Nevertheless, SQL expressions within

INSERT queries can still include built-in functions capable of returning critical information about the database, such as the database version, user details, and other metadata. Therefore, even without a WHERE clause, INSERT queries can still be vulnerable to certain types of injection attacks, necessitating the implementation of security measures to mitigate these risks.

Experimental part

In our experiments, we explored the security implications of SQL injection attacks within INSERT queries. Through a series of tests, we discovered that INSERT queries provide an opportunity to evaluate arbitrary SQL expressions via built-in functions, which can lead to the extraction of valuable information from the database. This capability becomes particularly critical when an attacker has root access, allowing them to access the server's file system.

For example, consider the following SQL expression:

$$if(user() = 'root@localhost', benchmark(999999, MD5(1)), 0)$$

This query utilizes the benchmark() function, which can be used to introduce a measurable delay in query execution. By observing the execution time, an attacker can determine whether the current database user is the root user (i.e., if the condition user()='root@localhost' is true). If this condition is met, the attacker gains the ability to invoke the LOAD_FILE() function, which grants access to the file system of the server. This access enables the extraction of potentially sensitive files or configuration data.

Additionally, if the username is not a standard name (e.g., not "root@localhost"), an attacker can employ character-by-character comparison techniques to retrieve the username using functions such as SUBSTRING() and ASCII(). This approach allows an attacker to systematically retrieve each character of the username, bypassing the need for any explicit error messages or database knowledge. Similarly, attackers can use these methods to extract values for other critical functions, such as DATABASE(), VERSION(), and LOAD_FILE().

To defend against such SQL injection attacks in INSERT queries, it is essential to sanitize and filter all data involved in generating the query. Specifically, special characters such as quotes (') and backslashes (\) must be properly escaped to prevent malicious input from altering the structure of the query. Additionally, implementing parameterized queries and using prepared statements can provide further protection by ensuring that user input is treated strictly as data, not executable code.

By taking these protective measures, database administrators can significantly reduce the risk of exploitation through SQL injection in INSERT queries, thus securing sensitive data and preventing unauthorized access to critical resources.

Conclusion

In conclusion, our research has demonstrated that SQL injections in INSERT queries represent a viable attack vector, capable of extracting sensitive information about the database and, in some cases, enabling unauthorized access to the file system of the attacked server. This finding underscores the importance of recognizing the security risks associated with INSERT queries in web application development. To mitigate such threats, it is essential to implement thorough input data validation and filtering mechanisms across all types of SQL queries. By adopting secure coding practices, such as parameterized queries and proper escaping of special characters, developers can significantly reduce the risk of successful SQL injection attacks, thereby enhancing the overall security of web applications.

References:

1. Normatov I., Yarashov I., Tangriberdiyev O. RESEARCH OF INTELLIGENT SYSTEMS FOR PROTECTION AGAINST NETWORK ATTACKs //Solution of social problems in management and economy. – 2023. – T. 2. – №. 11. – C. 128-136.
2. Normatov I., Yarashov I., Tangriberdiyev O. ON THE CONCEPT OF CREATING INTELLIGENT INFORMATION SECURITY SYSTEMS BASED ON NEURAL NETWORK INTRUSION DETECTION SYSTEMS //Science and innovation in the education system. – 2023. – T. 2. – №. 11. – C. 68-74.
3. Yarashov I., Shukurov D., Xudoyqulov K. REALIZATION OF ECONOMIC AND MATHEMATICAL MODELING OF INFORMATION SYSTEMS //CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES. – 2023. – T. 4. – №. 10. – C. 17-28.
4. Normatov I., Yarashov I., Tangriberdiyev O. Application of intellectual analysis to protect information in corporate systems //Central Asian journal of mathematical theory and computer sciences. – 2023. – T. 4. – №. 9. – C. 50-57.
5. Jumaniyozov Z. G. et al. Checking the condition of the shutter in the water distribution system using a laser sensor //Science and Education. – 2023. – T. 4. – №. 6. – C. 430-435.
6. Normatov I., Yarashov I., Boboqulov B. Development of models for describing the processing of environmental information in security problems of controlling a protection system based on Petri nets //Central Asian journal of

mathematical theory and computer sciences. – 2022. – Т. 3. – №. 12. – С. 229-239.
7. Kabulov A., Yarashov I., Daniyarov B. Systematic analysis of blockchain data storage and sharing technology //Central Asian journal of mathematical theory and computer sciences. – 2022. – Т. 3. – №. 12. – С. 240-247.
8. Бабаджанов А. Ф. и др. Алгоритмический анализ системы защиты информации на основе таблиц функционирования //International Journal of Contemporary Scientific and Technical Research. – 2022. – С. 216-219.
9. Normatov I. et al. Construction of reliable well distribution functions based on the principle of invariance for convenient user access control //2022 International Conference on Information Science and Communications Technologies (ICISCT). – IEEE, 2022. – С. 1-5.
10. Toshmatov S. et al. Designing an algorithmic formalization of threat actions based on a Functioning table //2022 International Conference on Information Science and Communications Technologies (ICISCT). – IEEE, 2022. – С. 1-5.
11. Yarashov I. Development of a reliable method for grouping users in user access control based on a Functioning table //2022 International Conference on Information Science and Communications Technologies (ICISCT). – IEEE, 2022. – С. 1-5.
12. Kabulov A. et al. Using algorithmic modeling to control user access based on functioning table //2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS). – IEEE, 2022. – С. 1-5.
13. Kabulov A., Yarashov I., Otakhonov A. Algorithmic Analysis of the System Based on the Functioning Table and Information Security //2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS). – IEEE, 2022. – С. 1-5.
14. Yarashov I., Normatov I., Mamatov A. The structure of the ecological information processing database and its organization //International Conference on Multidimensional Research and Innovative Technological Analyses. – 2022. – С. 114-117.
15. Yarashov I., Normatov I., Mamatov A. Ecological information processing technologies and information security //International Conference on Multidimensional Research and Innovative Technological Analyses. – 2022. – С. 73-76.

16. Kabulov A., Yarashov I., Mirzataev S. Development of the implementation of IoT monitoring system based on Node-Red technology //Karakalpak Scientific Journal. – 2022. – T. 5. – №. 2. – C. 55-64.
17. Kabulov A., Kalandarov I., Yarashov I. Problems of algorithmization of control of complex systems based on functioning tables in dynamic control systems //2021 International Conference on Information Science and Communications Technologies (ICISCT). – IEEE, 2021. – C. 1-4.
18. Kabulov A., Yarashov I. Mathematical model of information processing in the ecological monitoring information system //2021 International Conference on Information Science and Communications Technologies (ICISCT). – IEEE, 2021. – C. 1-4.
19. Kabulov A. et al. Development of an algorithmic model and methods for managing production systems based on algebra over functioning tables //2021 International Conference on Information Science and Communications Technologies (ICISCT). – IEEE, 2021. – C. 1-4.
20. Yarashov I. Algorithmic Formalization Of User Access To The Ecological Monitoring Information System //2021 International Conference on Information Science and Communications Technologies (ICISCT). – IEEE, 2021. – C. 1-3.
21. Kabulov A. et al. Algorithmic method of security of the Internet of Things based on steganographic coding //2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS). – IEEE, 2021. – C. 1-5.