Analysis and Prevention of SQL Injection based on Web

Junfeng Li

Jincheng College, Sichuan University, School of computer and software, Chengdu 611731, Sichuan, China

Abstract: ***With the continuous development of the network, individual or collective data are constantly generated and stored in the computer. If more private information is leaked or stolen, it will bring serious harm to individuals, families and even society. How to protect personal privacy or the security of system applications has become a top priority. Hackers on the Internet use the attack principle of different means to crack or "steal" and other malicious acts. SQL injection can be seen everywhere now, and it is also a common attack method. Knowing its principle can help us effectively prevent malicious attacks in this part.***

Keywords: SQL injection, Web security, Defense measures.

1. INTRODUCTION

With the explosive development of the Internet, web related applications can be seen everywhere in life. People produce data all the time in their daily life, and these data are also kept in the relevant web data system. For example, some people can analyze the most common online shopping by stealing the information of items purchased by customers[1]. For example, if you buy a milk bottle, others may find that the customer has a child at home or will have a child soon. Because the experience and level of developers who write code are uneven, and a large number of people do not consider the problem of security when writing code, most developers' ideas are basically that they can run fast and meet the requirements of customers. Officially, since there are not a few such developers, it gives some unscrupulous people an opportunity to take advantage of it. MPAAGN combines meta-path-guided neighbor selection, attention-based aggregation, and sampling strategies from GraphSAGE to enhance graph neural network performance, which can be utilized in other domains requiring efficient and scalable graph-based learning, such as social network analysis and recommendation systems [2]. The innovation lies in incorporating dynamic state transitions and contextual awareness into the Markov model, which enhances predictive capabilities and can be adapted to any recommendation or sequence-based prediction system [3].

2. WHAT IS SQL INJECTION

The name SQL injection is generally known in the IT industry[4-6]. It is a malicious attack that web programs are now facing. It mainly executes malicious SQL statements. Perform some operations on the database through some SQL statements [7]. In a simple sentence, the SQL statement executed by the program is different from the SQL statement that the developer wants the program to execute, resulting in completely different operation results, and generally the results are serious [8-10].

2.1 Classification of SQL Injection (According to the Type of Input)

Numeric injection point: when the parameter entered by the user is numeric, such as page number, it will be injected when the value of the input page number is 1 and 1=2. However, this usually occurs in weakly typed languages, such as PHP and ASP, but in strongly typed languages, errors will be reported, leading to injection failure.

String injection: common SQL operations include adding, deleting, modifying, and querying databases by using the where condition as a condition constraint. For example, select * from table name where name='(here is the condition, which is usually entered by the user). If the user inputs' or 1=1 #', the original statement will become select * from table name where name='or 1=1 #'. Then the SQL statement after injection means to query all records, not the original meaning: query name is the information of the name entered by the user.

Of course, according to different classifications, other standards can also be used to classify. Here, I just give a simple example, not a detailed variety of classifications. The main purpose is to explain some forms of SQL injection.

2.2 Main Features of SQL Injection Attack

There are many kinds of attacks: experienced veterans usually manually adjust the attack parameters, so that the attack data cannot be enumerated, which leads to some traditional feature recognition and matching methods can not correctly identify the attack and is difficult to prevent.

The attack operation is simple: there are a lot of SQL injection tools online, whether paid or free, and they can be used quickly. Attackers can use these tools to attack or destroy the target web effectively, quickly and simply.

The harm caused by the attack is great: due to some defects of the web itself, whether it is the language or its own configuration, and less security conscious developers, most web systems are likely to be injected. Once successfully injected, the attacker can steal, modify or even delete the data of the entire web, and the consequences will be serious.

2.3 Hazards of SQL Injection

The most serious is the database information or data leakage: the database stores all kinds of user related information, and the information of the database itself. Hackers who get these data will make the database more vulnerable to attack.

Malicious modification of web pages: hackers may tamper with specific web pages by modifying some information in the database.

The website is hung with a Trojan horse to spread malicious software: hackers may modify the information in the database and embed links to websites with great threats.

Attackers maliciously operate the database: modify the configuration or data of the database, and the administrator account or permission of the database is modified, causing the database server to fail to work normally.

The server is remotely controlled by hackers: usually, hackers will leave the back door after the operation so that they can "visit" next time.

After hackers get the permissions of the database, they first export the full table, and then destroy the database, or even the server system, leading to the collapse of the website.

3. MAIN PREVENTION METHODS OF SQL INJECTION

Since the web occupies the vast majority of the Internet, I have listed some common defense measures against such serious hazards:

3.1 Blacklist and Whitelist Protection

The first common way to prevent SQL injection is to use blacklist and whitelist to filter the fields passed into the database. White list: This is defined by the developer, which is equivalent to a standard formulated by the developer: the program only accepts the data entered by the user that meets the developer's standard. Of course, it is also possible that when a user inputs a very complex information, but it is not malicious data, but the standard defined by the developer is determined to be malicious, resulting in the user being unable to input the correct information, so this method is more suitable for the case of less input by the user, and for the case of more input, other prevention methods can be used to resist the attack; Blacklist is also a standard formulated by developers. It is used to reject user input, but the efficiency of this standard is not high, because developers have limited ideas, but there are still many threatening characters, otherwise there would not be so many bugs. The original method of using the database does have some effects, but the attacker is also very smart. They will also come up with more ways to bypass such a defense line.

3.2 Combination of Regular Expression and Business Logic

Because the user's input is not strictly verified, the new prevention technology using regular expressions and business logic is a more effective verification method [, but regular expressions are not understood by ordinary people, and the requirements are high, and the matching strings are relatively limited, so it is possible to filter out the SQL statements without problems.

3.3 Honeypot System Model

A defense model based on honeypot system. Honeypot is a kind of security resource. Its mission is to become the target of unauthorized access or attack, so as to obtain the attacker's attack behavior and attack strategy []. This system is equivalent to a trap for malicious attackers. Make a fake or meaningless business vulnerability, let malicious attackers deliberately enter this trap, and then use the recorded log to analyze, and then filter these malicious SQL statements, or write them into the blacklist, which is equivalent to allowing the system to continuously learn and evolve, so as to shield malicious attacks. Moreover, normal users generally do not fall into such a false trap, which also enables users to use the business of the system normally.

3.4 Parameterized Value Transmission

Parameterizing the input query conditions: This method is currently widely used and is an effective defense method; For example:

```
SqlConnection conn = new SqlConnection(". . . ");
conn.Open();
SqlParameter a = new SqlParameter("@ id", id);
SqlCommand cmd = new SqlCommand ( sql, conn);
cmd.Parameters.Add(a); cmd.ExecuteScalar ();
conn.Close();
```

The meaning of the above code is to use @ id to first form a complete SQL statement, which can be imagined as SELECT * FROM table name WHERE ID='@ id'. After executing this sentence in the database, the user's input ID is compared with the previously queried data, eliminating the need to execute SQL statements. This avoids using user input information as SQL statements in the database, which can effectively prevent the possibility of injection and is currently one of the best choices. But no situation is absolute. Although it can withstand most attacks, it still cannot achieve a foolproof defense. Malicious attackers can use some packet capture tools to capture packets and then modify the parameters inside the packets. Not only that, some malicious attackers may insert attack code into HTML at locations where they interact with the database, so that the malicious code can be stored in the database. When users access other locations on the web, SQL injection occurs.

4. BINARY STORAGE DATA

Storing user input data in binary format: One of the reasons for successful SQL injection attacks is that the data retrieved from the database and the information entered by the user are both character based data without clear boundaries. Moreover, computers are always just machines, and it is impossible to distinguish between the two, whether the information input by the user is malicious or normal. Due to the inability of computers to distinguish, they follow the principle of executing any statement they recognize, thereby changing the logic of expected SQL statements; If binary is used to store user input data and database data, the database can clearly distinguish who is who when executing SQL statements, and the occurrence of SQL injection will be greatly reduced. There are also related methods in development languages to convert data into binary data, and they are relatively simple. However, storing binary data in a database can make the work of database maintenance personnel difficult, as people are still somewhat familiar with Chinese or strings.

4.1 Using Stored Procedures

It pre writes the statements to be executed and saves them in the database, which can be directly called by the program when needed. Using relatively secure stored procedures can increase the resistance to SQL injection, but if the filtering is not thorough enough, there may also be SQL injection situations.

4.2 Pre compilation principles for using databases

It mainly has two functions, one is to improve the efficiency of data queries. For example, when a certain SQL statement is relatively long, has complex functions, or needs to be repeatedly executed, only a pre compiled SQL statement needs to be created for database parsing, and parameters can be passed in when the program needs to call it. Secondly, it has improved security. The data input by the user is directly sent to the database in plain text format, eliminating the need to concatenate SQL in the code. This creates a separation between code and data, theoretically eliminating SQL injection.

4.3 Vulnerability Scanning

For company network administrators, using some vulnerability scanning tools (paid and free) to regularly scan the system is a way to timely discover web security vulnerabilities, and relevant personnel can take targeted measures to effectively avoid the risk of SQL injection. For example: SQLIer, SQLMap, SQLNinja, etc. I have used SQLMap before, and I have to say that its content is really rich. Just pay attention to the scanning speed when scanning, it cannot be too fast, because if the speed is too fast, the server may not be able to handle it and there may be time outs.

5. CONCLUSION

Due to the rapid development of the web, various web servers have insufficient vulnerabilities and programs, and malicious attack methods are constantly emerging. SQL injection is gradually becoming the mainstream web attack method. More and more websites are being attacked, causing serious damage to the interests of individuals or groups. As a developer, it is necessary to have a deeper understanding of the causes of SQL injection attacks or vulnerabilities, and carefully study the detection and prevention methods of SQL injection to fundamentally protect data from malicious infringement and theft.

REFERENCES

[1] Wu, Z. (2024). An Efficient Recommendation Model Based on Knowledge Graph Attention-Assisted Network (KGATAX). arXiv preprint arXiv:2409.15315.

[2] Wu, Z. (2024). MPGAAN: Effective and Efficient Heterogeneous Information Network Classification. Journal of Computer Science and Technology Studies, 6(4), 08-16.

[3] Wu, Z., Wang, X., Huang, S., Yang, H., & Ma, D. (2024). Research on Prediction Recommendation System Based on Improved Markov Model. Advances in Computer, Signals and Systems, 8(5), 87-97.

[4] Yan, H., Wang, Z., Xu, Z., Wang, Z., Wu, Z., & Lyu, R. (2024). Research on image super-resolution reconstruction mechanism based on convolutional neural network. arXiv preprint arXiv:2407.13211.

[5] Jiang, L., Yu, C., Wu, Z., & Wang, Y. (2024). Advanced AI framework for enhanced detection and assessment of abdominal trauma: Integrating 3D segmentation with 2D CNN and RNN models. arXiv preprint arXiv:2407.16165.

[6] Zhu, Z., Wang, Z., Wu, Z., Zhang, Y., & Bo, S. (2024). Adversarial for Sequential Recommendation Walking in the Multi-Latent Space. Applied Science and Biotechnology Journal for Advanced Research, 3(4), 1-9.

[7] Wu, X., Wu, Y., Li, X., Ye, Z., Gu, X., Wu, Z., & Yang, Y. (2024). Application of adaptive machine learning systems in heterogeneous data environments. Global Academic Frontiers, 2(3), 37-50.

[8] Shen, Z. (2023). Algorithm Optimization and Performance Improvement of Data Visualization Analysis Platform based on Artificial Intelligence. Frontiers in Computing and Intelligent Systems, 5(3), 14-17.

[9] Zheng Ren, "Balancing role contributions: a novel approach for role-oriented dialogue summarization," Proc. SPIE 13259, International Conference on Automation Control, Algorithm, and Intelligent Bionics (ACAIB 2024), 1325920 (4 September 2024); https://doi.org/10.1117/12.3039616

[10] Z. Ren, "Enhancing Seq2Seq Models for Role-Oriented Dialogue Summary Generation Through Adaptive Feature Weighting and Dynamic Statistical Conditioninge," 2024 6th International Conference on Communications, Information System and Computer Engineering (CISCE), Guangzhou, China, 2024, pp. 497-501, doi: 10.1109/CISCE62493.2024.10653360.