THE INTERNATIONAL JOURNAL OF SCIENCE & TECHNOLEDGE

Defeating SQL Injection on Preventing Run Time Attacks

P. Sai Prathap Pullagura
M.E Computer Science and Engineering, Jay Shriram Group of Institutions, India
A. Gokilavani
Assistant Professor, Jay Shriram Group of Institutions, India

Abstract:
Most information systems and business applications built nowadays have a web frontend. These web applications can be accessed from anywhere and it become so widely exposed, any existing security vulnerability will most probably be uncovered and exploited by hackers. Two of the most widely spread and critical web application vulnerabilities are SQL Injection and XSS. SQLi and XSS allow attackers to access unauthorized data 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 or benefit ratio compared to other types of attacks. The proposed system developed with the security level very much enhanced from its actual level. The DBA cannot view the user details in its original form. The hacker cannot enter the user login by using the tricky queries, cannot run the inbuilt function.

Key words: *SQL Injection and XSS, DBA*

1. Introduction

In the field of security, the popularization and application of Internet, communications and computer network technology has been rapid development, especially the emergence of the Internet, makes the computer used in government, business, business, education, health care and other areas of society at an unprecedented rate, which are profound impact on people's economic, work and live. Network brings you convenience while brings more and more malicious attackers. Attackers target the network, database; make the database information security under serious threat. The SQL attack is one of common attacks, the tool of the SQL attack is SQL statements. Attackers towards programming vulnerability of application developers, submit well constructed SQL statement to the server to achieve the goal of attacking. The main objective is to keep database in a well secured manner under serious SQL injection attacks and to analyze the principle of SQL attacks, since it is considered to be a serious attack on a database. It provides safety methods to both users as well as administrators. It also avoids the administrators bypassing the user accounts. Various Illegal functions such as deleting records from the database, adding unwanted information to the database, running the innermost function in the database are also can be eradicated using various counter measures that were implemented in project.

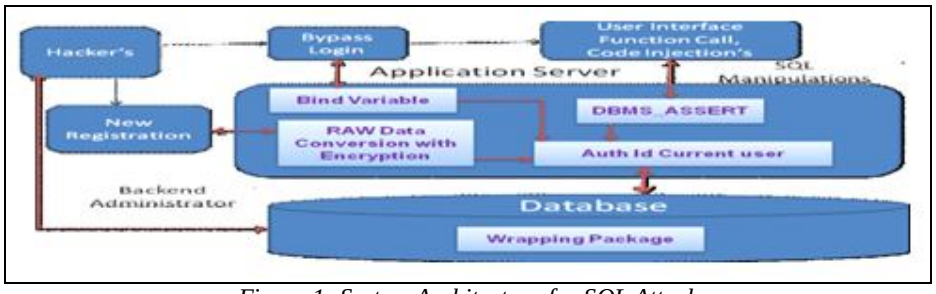


Figure 1: System Architecture for SQL Attack

2. Existing System

Network security practitioners put more resources and effort in to defending against SQL INJECTION ATTACKS, attacks, hackers will develop and deploy the next generation of SQLIA 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 information. It can easily execute oracle function or custom function from the select statement. If a DBA knows the user password, he can easily access user account without user permission.

2.1. Drawbacks of Existing System

- Bypasses the login authentication.
- Selects secure information from database tables.
- Attempts to add addition SQL statements to the existing SQL statement.
- Execute oracle function or custom function from the select statement.

3. Proposed System

The proposed system developed with the things that eradicate the drawbacks of the existing system. The security level very much enhanced from its actual level. The DBA cannot view the user details in its original form. The hacker cannot enter the user login by using the tricky queries, cannot run the inbuilt function in it etc.

3.1. Merits Of Proposed System

- Avoid unauthorized access to the application.
- User cannot get secure information from database.
- User cannot add sql statement to the existing SQL statement.
- User cannot call oracle function or custom function.

3.2. Problem Definition

A SQL injection attack involves the alteration of SQL statements that are used within a web application through the use of attacker-supplied data. Insufficient input validation and improper construction of SQL statements in web applications can expose them to SQL injection attacks. The hacker can retrieve the information of a user using various SQL injections such as tautology, piggybacked queries, union query, alternate encodings, malformed queries, inference, leveraging stored procedures, etc.

3.3. Problem Analysis

The purpose of the System Analysis is to produce the brief analysis task and also to establish complete information about the concept, behavior and other constraints such as performance measure and system optimization. The goal of System Analysis is to completely specify the technical details for the main concept in a concise and unambiguous manner. These problems can be analyzed from the point of a user through various techniques.

3.4. Packages Selected

The oracle platform provides a shared and elastically scalable platform for consolidation of existing applications and new application development and deployment. Oracle Database is an object relational database management system (ORDBMS). Oracle database conventions refer to defined groups of object ownership. As the system is to be developed in database security domain, we had preferred oracle platform with windows Application that supports all Networking components and procedures. Windows XP with all features is selected as the Development (Operating System) area to install and develop the system in oracle platform.

3.5. Resources Required

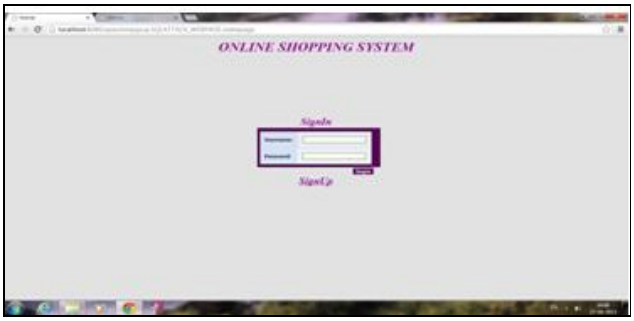
In this phase need to analyze the availability of the resources that are required to design, develop, Implement and Test the project. The resources to be analyzed are Manpower, Time and the system Requirements. Teams of two members are involved in the entire SDLC life cycle except the testing phase. The testing phase is guided by the professional testers before the implementation of the product. Time Analyzed to complete the project is approximately four months with 4 hrs on daily basis except weekends. System requirements are analyzed and listed below.

3.6. Feasiblity Study

The objective of feasibility study is not only to solve the problem but also to acquire a sense of its scope. During the study, the problem definition was crystallized and aspects of the problem to be included in the system are determined. Consequently benefits are estimated with greater accuracy at this stage. The key considerations are:

- Economic feasibility
- Technical feasibility
- Operational feasibility.

4. Results



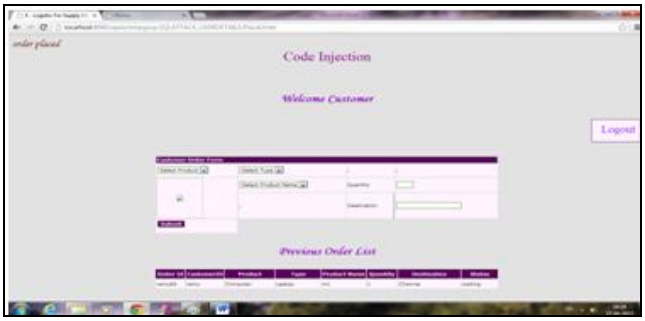


Figure 2: Home Page for SQL Attack
Figure 3: Order Placed by Customer in SQL Attack

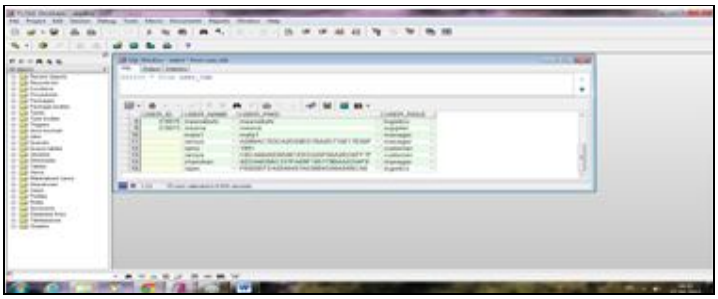


Figure 4: User Details in SQL Attack Free

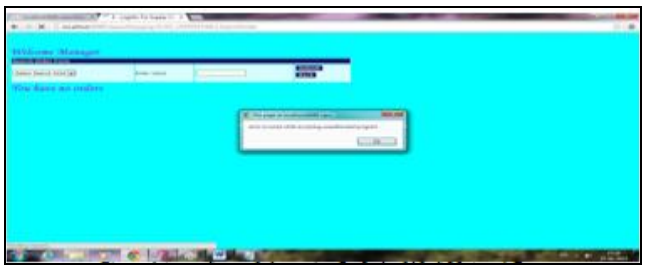


Figure 5: By Pass Login As Customer In SQL Attack Free
Figure 6: Access Secure Information By Order Id In SQL Attack Free

Figure 7: Emp_tab after Code Injection
Figure 8: Code Injection as Customer

5. Conclusion

The principles of SQL attacks and attack processes are analyzed. It introduces the visiting process based on client or server model. The proposed system developed with the things that eradicate the drawbacks of the existing system. The security level very much enhanced from its actual level. The DBA cannot view the user details in its original form. The hacker cannot enter the user login by using the tricky queries, cannot run the inbuilt function in it etc. On this basis, a database protection system against SQL attacks, mainly including the protection for ordinary users and administrators is achieved. Experiments show that this is a very effective protection system.

6. References

1. Andrew Wood, Kate Downer and Annalise Toberman, 2001**,** Department for Work and Pensions Research, "Evidence review of smartcard schemes in local authorities", pp. 738.
2. Chris Anley , 2002,vol 2, Next Generation Security Software, "Advanced SQL Injection In SQL Server Applications and (more) Advanced SQL Injection". Pp.253-254
3. Meixing Le, AngelosStavrou, and Brent ByungHoon Kang, 2012,Vol .9, Dependable and Secure Computing "Doudle Guard: Detecting Intrusions In Multitier Web Applications ", pp.77-78.
4. NurulZawiyahBinti Mohamad,2008,"Effectiveness Of Structured Query Language Injection Attacks Detection Mechanisms", pp.122-125.
5. Rahul Shrivastava, Joy Bhattacharyji, RoopaliSoni,2012, vol.2, Asian journal of Computer Science and Information Technology,"Sql injection attack in database using webservice: detection and prevention", pp.162-165.
6. F. Valeur, G. Vigna, C. Kru¨ gel, and R.A. Kemmerer, 2004,vol. 1, no. 3, IEEE Trans. Dependable and Secure Computing,"AComprehensive Approach toIntrusion Detection Alert Correlation," pp. 146-169.
7. G. Vigna, F. Valeur, D. Balzarotti, W.K. Robertson, C. Kruegel, and E. Kirda,2009, vol. 17, no. 3, J.ComputerSecurity,"Reducing Errors in the Anomaly-Based Detection ofWeb-Based Attacks through the Combined Analysis of Web Requests and SQL Queries", pp. 305-329.
8. William G. J. Halfond, Jeremy Viegas and Ro Orso, 2012, vol 7, "A Classification of SQL Injection Attacks and Countermeasures"pp.56-59