Detection and Avoidance of SQL Injection Attack In Multi-Tier Web Based Application

Ms. S.Raichal[2] ,Assistant Professor,[2]Email:s.raichal@gmail.com
Ms.Soniya[3] ,Assistant Professor, [3]Email:j.soniya0709@gmail.com
Ms. V.M.Gayathri[1] ,Assistant Professor,[1]Email:vmg188@gmail.com
Dr. R.Nedunchelian[4] , Professor & Head, [4]Email:chelian1959@gmail.com
(1),(2),(3),(4) Department of Computer Science & Engineering,
Saveetha school of engineering,Saveetha University Chennai-602 105.

Abstract:

The aim of this paper is to prevent sql injection attack using stored procedure. In SQL injection attack, an attacker might deliver malicious SQL query segments as user input which could effect in a different database request. Using SQL injection attacks, an attacker might thus obtain and modify confidential information. An attacker could even use a SQL injection vulnerability as a basic IP or Port scanner of the internal corporate network. The stored procedure does not permit conditional statement there by the hacker cannot identify the IDs. The stored procedure is the new approach that is executed. Stored procedure avoids the attack which is more secured one where the conditional statements are not permitted. In sql injection input is set as conditional statements and the user can able to login into the website. But in this paper every condition are checked in the procedural language of stored procedure. Once it notices the condition statement the user will be blocked to log into the website. Only the correct form of passwords is acceptable.

Keywords: SQL injection, stored procedure, conditional statement.

solicitation

I INTRODUCTION

The attacks have recently become more varied, as attention has shifted from attacking the front end to exploiting vulnerabilities of the web applications in order to corrupt the back-end database structure (e.g., SQL injection attacks) the back-end database server is often safe behind a firewall while the web servers are remotely accessible over the Internet. Inappropriately, though they are protected from direct remote attacks, the back-end systems are vulnerable to attacks that use web requests as a means to exploit the back end.

Server attacks are measured to be more server in Major server level can be attacked through SQL Injection Attacks. The aim is to search out attacks that independent IDS would not be able to identify by using stored procedures .Internet services have been soon flourished across the globe in relation to its popularity and complexity. The World Wide Web provides varied applications such as the social media, educational, finance etc. But, the wide use of these services has made them exist in on the peak of attackers. Different types of Attacks [1] like SQL injection and hijacking period of users are carried out on the web servers here the webserver is taken over by the attacker. Hence, all the subsequent user sessions are also being hijacked.

The anomaly detection systems [2] detect the irregular behaviors of the system. The network packets are individually analyzed by the intrusion detection systems. But for the system considered to be multitier a very modest task is carried out. Hence, due to the absence of such architecture even if we make use of firewall to guard the database backend, they are likely to be attacked by the attackers who employ the web request for misusing the back end data. The misused detection [3] can be used for matching the changed patterns in order to prevent the multi tier web services. But, when the network traffic is transferal from the web to the database or vice-versa, the intrusion detection system is unable to identify the anomalous traffic by just using the web intrusion detection system or the database intrusion detection system.

For ill-using the vulnerabilities in the web, the attacker uses the non admin rights by delivering a privileged database query. This type of attack can neither be discovered by the database intrusion detection system nor by the web intrusion detection system. It is be impossible to detect the unintended mapping between the database and the web server with the current web server multithreaded architecture.

Here, a solution to the problem which deals with the intrusion detection and prevention in the multitier web architecture is offered. The multitier web application analyser is being advanced for preventing the system from wide number of attacks. Normality Model of the isolated user sessions is being made which consists of the

web application front end that is in form of HTTP requests and the data base backend [4] i.e. the SQL or file server. The intrusion prevention system builds up a casual mapping model which is to be done by taking under attention both the web server and database traffic.

The intrusion prevention system is the conservatoire of intrusion detection system. In the intrusion prevention system, the susceptibilities are being blocked and therefore do not affect the system or network any longer. Here, presenting a lightweight virtualization model for assigning every user's session to a container. To specifically relate the web request with the resulting database queries we formulate use of the container ID. The IPS forms a casual mapping profile by undertaking the web server and database traffic. OpezVZ [5] is used for implementing the IPS container architecture. This container based architecture along with casual mapping also provides an isolation which helps in preservating the session hijacking attacks of the users.

The lightweight virtualization [6] helps in running innumerous replicas of the web server instances in different containers. Hence, everyone is separated from the rest. Each client is assigned a dedicated container so that even when the attacker attacks the session, it is limited to that session only and doesn't harm the other user session.

In SQL injection, the attacker will give the password as a conditional statement which will confuse the database ,so that the attacker may enter into the login of many users(for ex: if the attacker enters into the shopping website he may purchase in different user ID). So, Stored Procedure is to avoid this kind of attack which is more secured one were the conditional statements are not allowed.

II SYSTEM STUDY

2.1 Existing system:

In existing system this sort of web vulnerabilities cannot be identified that easily since there is a victim affected by it. Only protection measures are handled for this sort of attacks. Attackers can cause injuries to both servers as well as end users, but the motivation is to disturb the end users only. Either if Servers and End users are not attentive of the attackers then there is a definite loss. Most of the end users are not aware of the kind of attackers who are accessible through the web. Attacker can give the password as a conditional statement which complicate the backend database in turn the attacker may enter into the login of many users. For ex: if the attacker enters into the shopping website he may purchase in diverse user ID.

A network Intrusion Detection System can be categorized into two types: anomaly detection and misuse detection. Anomaly detection first needs the IDS to define and describe the correct and acceptable static form and dynamic behavior of the system, which can then be used to detect abnormal variations or anomalous behaviors. Intrusion alerts correlation provides a collection of components that convert intrusion detection sensor alerts into succinct intrusion reports in order to reduce the number of replicated alerts, false positives, and non relevant positives.

2.2 Proposed system:

The proposed system not only handles the safeguard measures but also the counter measures. The proposed system was designed to afford counter measures for such attacks namely Denial of Service (DoS) Attack, SQL Injection (Database Attack). Monitoring process is considered for DoS and SQL Injection attack. The stored procedure is the new method that is implemented. Stored procedure avoids the attack which is more secured one where the conditional statements are not allowable.

This type of attack can be readily identified if the database IDS can identify that a privileged request from the web server is not associated with user-privileged access. Unfortunately, within the current multi threaded web server architecture, it is not possible to detect or profile such causal mapping between web server traffic and DB server traffic since traffic cannot be visibly attributed to user sessions.

Stored procedures assist in achieving a constant implementation of logic across applications. The SQL statements and logic needed to perform a normally performed task can be designed, coded, and tested once in a stored procedure. Each application needing to perform that task can then simply perform the stored procedure. Coding business logic into a single stored procedure also offers a single point of control for safeguarding that business rules are correctly enforced.

Stored procedures can also increase performance. Many tasks are implemented as a sequence of SQL statements. Conditional logic applied to the results of the first SQL statements determines which successive SQL statements are executed. If these SQL statements and conditional logic are written into a stored procedure, they become portion of a single execution plan on the server. The results do not have to be returned to the client to have the conditional logic applied; all the work is done on the server.

SQL injection is an attack procedure that targets the data residing in a database. The attack takes gain of poor input validation in code and website administration. SQL Injection Attacks occur when an attacker is able to insert a sequence of SQL statements into a 'query' by manipulating user input data into a web-based application, an attacker can take advantages of web application programming security faults and pass

unexpected malicious SQL statements through a web application for execution by the back-end database. This paper suggests a novel specification-based methodology for the prevention of SQL injection Attacks.

2.3 Feasibility study:

All projects are achievable given unlimited resources and infinite time. It is both necessary and prudent to calculate the feasibility of the project at the earliest possible time. Feasibility and risk analysis is related in many ways. If project risk is great, the feasibility listed below is equally important. The following feasibility techniques have been used in this project.

There are three feasibility and they are

- Operational Feasibility
- Technical Feasibility
- Economic Feasibility

2.3.1 Operational Feasibility:

Proposed system is beneficial since it turned into information system analyzing the traffic that will meet the organizations operating requirements. In security, the file is transferred to the destination and the acknowledgement is given to the server. Bulk of data transfer is sent without traffic.

2.3.2 Technical Feasibility:

Technical feasibility centers on the existing computer system (hardware, software, etc...) and to what extent it can support the proposed addition. For example, if the current computer is operating at 80% capacity. This involves, additional hardware (RAM and PROCESSOR) will rise the speed of the process software and normal hardware configuration is enough, so the system is more possible on this criteria.

2.3.3 Economic Feasibility:

Economic feasibility is the most frequently used method for valuing the effectiveness of a candidate system. More commonly known as cost / benefit analysis, the procedure is to determine the benefits and saving that are expected from a candidate and compare them with the costs. If the benefits outweigh cost, then the result is made to design and implement the system. Otherwise drop the system.

This system has been executed such that it can be used to analysis the traffic. So it does not require any extra equipment or hardware to implement. So it is economically feasible to use.

III SYSTEM IMPLEMENTATION

3.1 Phases:

- Server Deployment
- Server Vulnerability and Attack
- Monitor and Recovery

3.1.1 Server Deployment:

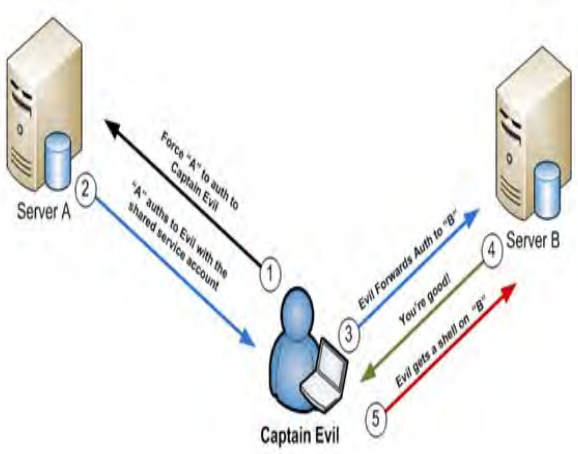
Websites are planned to start a connection between the administrators (website owners) and the end users. Here the end users are the common people. When these websites are hosted over the internet they can be accessed from all over the world unrelatedly of the country and continent. This is more useful for a business minded people to get clients across globally.

To prove the vulnerability of the website and its rectification methods so ,its needed a website for this purpose . No third party website can be usaed for this purpose. So creating a website for using asp.net. and to deploy this website so that it could be accessible over the LAN network. A application server like IIS or Apache Tomcat for this purpose. Developing a application server and deploy the website that we have created. So the website created looks like the miniature model of that of the website that was created.

3.1.2 Server Vulnerability and Attack:

This module explains about the server side attacks.

The server side attacks are the server weakness. The two different types of attack that affects the server.The website is designed in the format of three tier architecture.



This tier consists of the application server, end user the database server They form the three tier of the website. Server vulnerability on the application server are affected by DoS(Denial of Service) and Sql Injection Attack. These attacks show that the servers are vulnerable in nature.

3.1.3 Monitor and Recovery:

The monitor and recovery is the phase that is presented in order to avoid this Server vulnerability. The monitoring and recovery can be considered into two different phases that avoids the DoS attack and Sql Injection Attack.

A stored procedure is an operation set that is stored. Normally, stored procedures are written in SQL. Since stored procedures are stored on the server side, they are available to all clients. Once the stored procedure is altered, all clients automatically get the new version.

Security Monitoring and Attack Detection requires planning the suitable levels of security audits for the following areas:

- Account management
- Protected file access
- Security policy changes
- Trust creation and deletion
- User rights usage
- System restarts and time changes
- Registry modifications
- Unknown program execution

The security monitoring and attack detection system collects information from the security event logs and assembles this information in a central location. Security auditors can then analyze this data for suspicious activity.

```
1. CREATE PROCEDURE [EMP].[RetrieveProfile] @Name varchar(50),
@Passwd varchar(50)
2. WITH EXECUTE AS CALLER
3. AS
4. BEGIN
5. DECLARE @SQL varchar(200);
6. ...
7. SET @SQL='select PROFILE from EMPLOYEE where ';
8. ...
9. IF LEN(@Name) > 0 AND LEN(@Passwd) > 0
10. BEGIN
11. ...
12. SELECT @SQL=@SQL+'NAME='''+@Name+''' and ';
13. SELECT @SQL=@SQL+'PASSWD='''+@Passwd+'''';
```

```
14. ...
15. END
16. ELSE
17. BEGIN
18. ...
19. SELECT @SQL=@SQL+'NAME="Guest"';
20. ...
21. END
22. ...
23. EXEC(@SQL)
24. ...
25. END
```

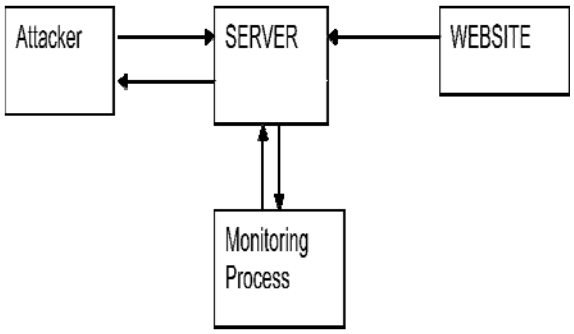
CODE : Stored Procedure vulnerable to SQL-Injection

The monitoring process monitors the application Server in such a way that Number clients are using the particular website and its packet counts. If the application server receives a irregular mount of packets from any of the end users the servers identifies it as the Denial of Service(DoS) attack and the countermeasures are taken by the server in such a way that the server terminates the misbehaviors.

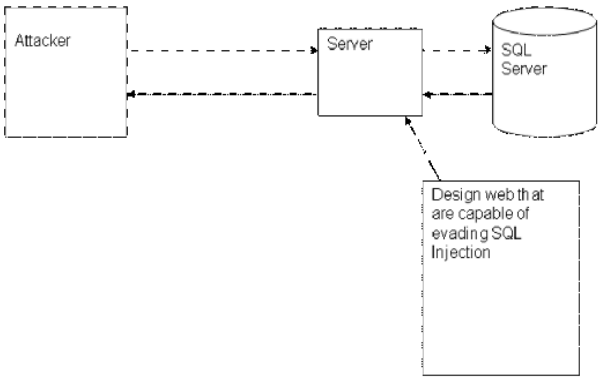
IV SYSTEM DESIGN

The systems architect establishes the basic structure of the system, defining the essential core design features and elements that provide the framework for all that follows, and are the hardest to change later. The systems architect provides the architects view of the users' vision for what the system needs to be and do, and the paths along which it must be able to evolve, and strives to maintain the integrity of that vision as it evolves during detailed design and implementation.

Attack 1:



Attack 2:



V EVALUATION

This section shows the results of experimental analysis. The main advantage of proposed method is its simplicity. In this type of attack an attacker can operate white spaces to prevent an attack from detection. It presents a novel technique to dynamically de- duce the programmer intended structure of SQL queries and used it to successfully convert applications so that they guard themselves against SQL injection attacks.

VI CONCLUSION

The most important attack scenarios are focused as DoS and SQL Injection attack. Once the database is conceded, the attacker can grab searching data, such as usernames, passwords, credit card info, etc., or they can add data to your system to do things like redirect your legitimate website viewers to spyware websites. This project plays the important role in detecting and preventing the Sql injection attack. The attacks like the SQL injection make use of the database queries with injection to database server. So there is a need to defend these systems in the cloud storage. It offers a better characterization of the system for anomaly detection because the intrusion sensor has a more precise normality model that detects a wider range of threats.

REFERENCES

[1] C. Anley, “Advanced Sql Injection in Sql Server applications,” technical report, Next Generation Security Software, Ltd., 2002.
[2] H. Debar, M. Dacier, and A. Wespi, “Towards Taxonomy of Intrusion-Detection Systems,” Computer Networks, vol. 31, no. 9, pp. 805-822, 1999.
[3] Liang and Sekar, “Fast and Automated Generation of Attack Signatures: A Basis for Building Self-Protecting Servers,” SIGSAC: Proc. 12th ACM Conf. Computer and Comm. Security, 2005
[4] Y. Shin, L. Williams, and T. Xie, “SQLUnitgen: Test CaseGeneration for SQL Injection Detection,” technical report, Dept.of Computer Science, North Carolina State Univ., 2006.
[5] G. Li, B.C. Ooi, J. Feng, J. Wang, and L. Zhou, “Ease: An Effective 3-in-1 Keyword Search Method for Unstructured, Semi-Structured and Structured Data,” Proc. ACM SIGMOD Int’l Conf. Management of Data, pp. 903-914, 2008.
[6] C. Anley. Advanced sql injection in sql server applications. http://www.nextgenss.com/papers/advanced sql injection.pdf.
[7] C. Anley. (more) advanced sql injection. http://www.nextgenss.com/papers/more advanced sql injection.pdf, White Paper.
[8] B. M. L. Archive. http://seclists.org/lists/bugtraq/2005/,2005