Prevention and Detection Techniques for SQL Injection Attacks

Kamlesh Kumar Raghuvanshi [#1], Deen Bandhu Dixit [#2],

[#1]*Asst. Professor(Adhoc) PGDAV College, Delhi University ,New Delhi India*

[*2]*Department of Computer Science, New Delhi , India*

Abstract— **:** Different kind of Web Application run with help of web server using World Wide Web protocol and many of web application could be vulnerable with SQL Injection attacks this is the type of input validation attacks. Using this type of attacks, web application could be hacked easily and steal the confidential data by the anonymous user. This may be dangerous for organization market value.

Keyword: **SQLInjection Attacks, Static Analysis, Dynamic analysis, Detection, Prevention**

I. Introduction

SQL Injection Attacks is the effective mechanism for hacker for stealing the useful information regarding web application or database management system, use of this attack anonymous user obtain the confidential information. Basically such attacks are obtained through web input so that SQL Injection attacks belongs to the category of input validation attack over the web application. Mostly web application hacked through SQL Injection Attacks method. It comes under top ten security threat in web applications [1]. Anonymous user obtained the different type of information with help of SQL injection type
.

II. Type of SQL injections ATTACKS

Following main categories of SQL Injection attacks against database server which exist on the web.

SQL Manipulation: It is the procedure of altering the SQL statement with the aid of built in operation or built in function like UNION. On the other way where clause can be sued for injecting the vulnerable SQL command this is also the method belong to the SQL manipulation category.

Code Injection: Code injection is the way of inserting vulnerable SQL statement. One o the code injection attacks is

Function Call Injection: with the help of this categories anonymous user could be insert SQL function call into a vulnerable SQL statement. Such type of injection could call the operating system procedure or modify data in backend.

Buffer Overflows: Buffer overflow is caused by using function call injection. For most of the commercial and open source databases, patches are available. This type of attack is possible when the server is un-patched.

III. Type of attacks

. There are several types of attack [2,3]. This paper represents various attacks with example.

Types of Attacks	**Description**
Tautologies	This attack belong to the SQL Manipulation category. Anonymous user can inject vulnerable code into SQL statement. This based upon conditional statement
Logically Incorrect Queries	This type of attack also belong to the SQL Manipulation category with the help of this attack anonymous user can get help message from error message which will produced by data base server.
Union Query	This attack belong to the code injection category and SQL manipulation category in which inject infected queries with safe queries to get table or database information.
Stored Procedure	This type of attack belong to the function call injection category. In this type attacks anonymous user execute built in procedure for getting table information or database information.
Piggy Backed Query Attack	This type of attack represent to the code injection category.
Alternate Encoding	In this type of attacks, SQL queries will be manipulate using alternate encoding, Hexadecimal, ACSII and Unicode.
Blind Attacks	In this type of attacks anonymous user can theft useful information from database asking true and false a question through SQL statement.

Stored XSS: It stands for Type-2 XSS it arises when web based application accepts hostile data, stores it in a file, database, or then back- closing system, and then large stage Displays that unfiltered information to the dupe. This is exceedingly dangerous in systems like blogs or forums and capacity management systems, where the figure of users sees input from the people. Even worse, if a utilizer with administrative privileges views an infected page or log ingression, the assailant could potentially hijack an administrative session.

DOM based cross site scripting Attacks: it is also some times called "type-0 XSS". It transpires when the XSS vector executes as a consequence of a DOM modification on a website in a user's browser. On the customer side, the HTTP replication does not transmute, butt the script executes in a maleficent way. This is the most advanced and least-recognized type of XSS. Most of the fourth dimension, this susceptibility subsists because developers do not visually perceive how it operates.

Web site can additionally be the conveyance of an assailment on your users by a third party, potentially making you appear to be the assailant. For instance, an attacker's site sends a ostensibly trustworthy link to a utilizer in an electronic-mail message The user clicks the link. All the same, the URL includes a parameter that transmits the user's name to the host so that the host can display that name to the user in some text string. The attacker has appended script tags and some malicious script to the URL, rather than a figure using the same type of codification.

IV. DETECTION AND PREVENTION TECHNIQUES

There is many ways to prevent and detect SQL Injection attacks. Prevention means faultlessness of user supplied input value at programming level. Such type of techniques enforce to the user to provide correct information and can be block irregular value which is unsafe to database server. That type of prevention done at both side it may be the client side or the server side. Input validation attacks like SQL injection may or may not defend against SQL Injection attacks. Following techniques for preventing and detecting SQL Injection attacks.

Ke wei, et al. [4] provide stored procedure approach with the aid of built in subroutines such subroutines is usually used for performing operation on database server. They detect SQL Injection attack using the combination concept of static and dynamic analysis for instruction verification by using SQLCHECKER [5] for input validation.

S. W. Boyd et al.[6] in this approach proxy server concept is used for de-cphering the queries between web server and server which is passed by the anonymous user or normal user. This task used two mechanism one is de randomize the SQL queries and forward queries with the standards set of keywords to database server for computation. And also it prevent the error message to seen by anonymous user or normal user correspond to the illegal queries.

William G.J. Halfond et al[7]. SQL injection detect with the aid of this approach which occurred over the web application environment this techniques belong to the static approach and also runtime monitoring. Model based approach are used for detect malicious injected code before executing on the database server it have two part static part which automatically builds a legal queries using programs analysis on the other hand in dynamic part it dynamically generates the queries against statically build queries using runtime monitoring. If supplied queries harm the techniques then it prevent the execution of the queries on the database server.

G. Buehrer et al[8]. in this approach comparison at runtime between parse tree of generated statement and original statement, execution is not start until the parse tree of the intended SQL query and the resulting SQL query generated after attacker input do not match.

R.A. McClure et al[9]. it is used CLI (call level interface) mechanism is used between web application and database like ODBC, JDBC and SQLCLIENT. This is the database provider used in language which contain strong type class to database. This mechanism used to SQLdomgen which used against database it force to the user write the correct code which is checked at compile time.

P.Bisht et al[10]. this approach to dynamic candidate evaluations which automatically prevent the SQL injection attacks is possible. This approach support to the dynamic construction of structure of intended queries whenever the issues the queries. This approach check the user passed value by running the application over the user input those represent to self-evidently non-attacking.

.

Shaukat Ali et al[11]. in this approach hash function used to improve performance of authentication for web based application. Username and password is created at runtime when user account is created then this hash value is used.

Takeshi Matsuda et al[12]. in this techniques SQL injection attacks is detected using to sigmoid function based on single character. The identification of attack character string when the character string is matched with SQL injection attack. This help to minimize the predictive error in SQL injection attack detection.

Angelo Ciampa et al[13]. Heuristic based approach is used for detect the SQL injection attack over the web application. It generally used panetration testing of web application. In this approach is help to determine web application hyper link structure and input given by the user and gives error message if SQL injection occurred in user supplied value.

Mei Junjin et al[14]. This is the category of manual approach that is two type of manual approach one is defencive programming which means programmer can write a code in such a way which restrict to the user that does not enter malicious code and keywords. And also programmer can be used to safe API for preventing SQL injection attack over the web application like SQL DOM approach.

Z. Su et al[15]. SQL APPROACH is used for check on given input queries with defined one by the developer and this approach applies secret key for user input determination. Advantage of this approach is there is no false positive or false negative. This also reduced runtime overhead and implemented in various platform which is used for developing the web application.

YongJoon Park et al[16]. In this approach Needalman-wunsch algorithm[17] for identify the sequential feature. This approach analyze the client web request data to normal request patterns and automatically generate profile of HTTP queries which is generate by the application. The advantage of this approach reduced the false positive rate and improves the performance.

G. Wassermann et al[18]. SQL Injection attack which type of tautology prevent through static analysis of input web request. This techniques is effective for finding out other SQLIA.

A. Nguyen-Tuong et al[19]. This is automatic approach for detecting and preventing input validation attacks over the web application and they used modified interpreter for detecting SQL Injection attacks that is tainted user supplied web request. This is also effective to prevent the generation of scripting code with the help untrusted input.

Adam Kie˙zun et al[22]. This approach support to indentify the SQL Injection attacks as well as XSS attacks vulnerabilities. This approach used on unmodified existing code, produced concrete input that expose attacks malicious and it operate before software sis deployed. It as an automated process for creating attacks. It require source code for analysis so that it is belong to the category white box testing. This approach is based on the input generation, taint propagation and input mutation to find variant of an execution that exploit vulnerabilities.

Raju haldar et al[21]. In this approach SQL injection attacks detect with the implementation of combine structure of static and dynamic analysis which is based on obfuscation de-obfuscation of SQL commands. SQLIA detection in easy way based on dynamic verification on the obfuscated queries. With the help of this techniques root cause of the SQL injection attacks find in dynamic query generator.

Shikhar Jain et al[22]. With the help static analysis approach of queries that is validation process of queries which generate at runtime. If any malicious code found in the generated queries those queries will not match with defined static queries thus such queries mark as attack and rejected. And different language use the tools for performing or support this type of approach like Dot Net String Analyzer(DNSA) and SDMGV(Static Dynamic Model Generator and validator).

Yao-Wen et al[23]. It is category of automation tool which uses static analysis to verify taint flows against sensitive procedure. In this process run time is active when any vulnerabilities occurred. Automated means this process executed without help of developer.

William G.J. Halfond et al.[24]. This approach use the concept of positive tainting , accurate and efficient taint propagation and syntax-aware evaluation of queries string and minimal deployment requirement.

T.Pietraszek eta al[25]. This is also automated techniques which help to prevent and detect SQL injection attacks which is type of the input validation attacks. This techniques uses a combination of assignment of metadata to user input and metadata preserving string operation and context sensitive string evaluation for detecting the SQL injection attacks. This CSSE(Context Sensitive String Evaluation) prototype support to PHP language.

Michael Martin et al[26]. In this approach use of the static and dynamic checker to prevent SQL injection. This techniques support to JAVA language. With the help of static checker find the all potential matches in an application and a dynamic checker tracks all matches precisely as they occur. This approach also maintain logging information and recovery action upon a match.

V. Conclusion

This paper represent the brief survey of the SQL injection attacks and also define the procedure of attack implement on the database server suing web input request inform SQL queries. And also describe the different SQL injection detection and prevention techniques listed. This help to researcher to create more effective detection and prevention system with the help this survey paper.

References

[1] OWASPD-Open Web Application Security Project. “Top ten most critical Web OWASPD-Open Web Application Security Project. “Top ten most critical Web Application Security Risks”, https://www.owasp.org/index.php/Top_10_2010-Main.

[2] W. G. Halfond, J. Viegas, and A. Orso. A Classification of SQLInjection Attacks and Countermeasures. In Proc. of the Intl. Symposium on Secure Software Engineering, Mar. 2006.

[3] Atefeh Tajpour, Suthaimi, Maslin Masrom. SQL Injection Detection and Prevention Techniques .In Proc. International Journal of Advancements in Computing Technology Volume 3, Number 7, August 2011.

[4] Ke Wei, M. Muthuprasanna, Suraj Kothari , “Preventing SQL Injection Attacks in Stored Procedures” Proceedings of the 2006 Australian Software Engineering Conference (ASWEC’06 IEEE).

[5] Z. Su and G. Wassermann “The essence of command injection attacks in web applications”. In ACM Symposium on Principles of Programming Languages (POPL’2006), January 2006.

[6] S. W. Boyd and A. D. Keromytis. SQLrand: Preventing SQL Injection Attacks. In Proceedings of the 2nd Applied Cryptography and Network Security Conference, pages 292–302, June 2004.

[7] William G.J. Halfond and Alessandro Orso,” Preventing SQL Injection Attacks Using AMNESIA” ICSE’06, May 20–28, 2006, Shanghai, China ACM 06/0005.

[8] G. Buehrer, B.W. Weide, P.A.G. Sivilotti, Using Parse Tree Validation to Prevent SQL Injection Attacks, in: 5th International Workshop on Software Engineering and Middleware, Lisbon,Portugal, 2005, pp. 106–113.

[9] R.A. McClure, and I.H. Kruger, "SQL DOM: compile time checking of dynamic SQL statements," Software Engineering, 2005. ICSE 2005. Proceedings. 27th International Conference on, pp. 88- 96, 15-21 May 2005.

[10] P. Bisht, P. Madhusudan, and V. N. Venkatakrishnan. CANDID: Dynamic Candidate Evaluations for Automatic Prevention of SQL Injection Attacks. ACM Trans. Inf. Syst. Secur., 13(2):1–39, 2010.

[11] Shaukat Ali, Azhar Rauf, Huma Javed. SQLIPA: An Authentication Mechanism Against SQL Injection. In Proc. European Journal of Scientific Research ISSN 1450-216X Vol.38 No.4 (2009), pp 604-611.

[12] Takeshi Matsuda,Daiki Koizumi,Michio Sonoda,Shigeichi Hirasawa, ”On predictive errors of SQL injection attack detection by the feature of the single character” Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on 9-12 Oct 2011, On Page 1722-1727.

[13] Angelo Ciampa, Corrado Aaron Visaggio, Massimiliano Di Penta :”A heuristic-based approach for detecting SQL-injection vulnerabilities in Web applications”.

[14] Mei Junjin, "An Approach for SQL Injection Vulnerability Detection," Proc. of ITNG '09, pp.1411-1414, 27-29 April 2009.

[15] Z. Su and G. Wassermann “The essence of command injection attacks in web applications”. In ACM Symposium on Principles of Programming Languages (POPL’2006), January 2006).

[16] YongJoon Park,JaeChul Park,”Web Application Intrusion Detection System for Input Validation Attack ”Third 2008 International Conference on Convergence And Hybrid Information Technology.

[17] Needleman, S.B., Wunsch, C.D. “A general method applicable to the search for similarities in the amino acid sequence of two proteins”, J.Mol.Biol.48:443-453, 1970.

[18] G. Wassermann and Z. Su. 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.

[19] A. Nguyen-Tuong, S. Guarnieri, D. Greene, J. Shirley, and D. Evans. Automatically Hardening Web Applications Using Precise Tainting. Security and Privacy in the Age of Ubiquitous Computing, Volume: 181, Pages: 295-307, 2005.

[20] Adam Kie˙zun,Philip J. Guo,Karthick Jayaraman,Michael D. Ernst:“Automatic Creation of SQL Injection and Cross-Site Scripting AttackS”, ICSE‟09, May 16-24, 2009, Vancouver, Canada,978-1-4244-3452-7/09/$25.00 © 2009 IEEE.

[21] Raju Halder and Agostino Cortesi, “Obfuscation-based Analysis of SQL Injection Attacks”. 978-1-4244-7755-5/10/$26.00 ©2010 IEEE

[22] Shikhar Jain & Alwyn R. Pais,” Model Based Approach to Prevent SQL Injection Attacks on .NET Applications” International Journal of Computer Science & Informatics, Volume-I, Issue-II, 2011.

[23] Yao-Wan Huang, Fang Yu, Christian Hang, Chung-Hung Tsai, D.T.Lee, Sy-Yen Kuo.Securing WebApplication Code by Static Analysis And Runtime Protection.In Proceeding of the 12th International World Wide Web Conference(WWW-04), May 2004.

[24] William G.J. Halfond, Alessandro Orso.” WASP: Protecting Web Applications Using Positive Tainting and Syntax-Aware Evaluation” IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 34, NO. 1, JANUARY/FEBRUARY 2008.

[25] T. Pietraszek and C. V. Berghe. Defending against Injection Attacks through Context-Sensitive String evaluation. Recent Advances in Intrusion Detection, Volume: 3858, Pages: 124-145, 2006.

[26] M. Martin, B. Livshits, and M. S. Lam. Finding Application Errors and Security Flaws Using PQL:A Program Query Language. ACM SIGPLAN Notices, Volume: 40, Issue: 10 Pages: 365-383,2005.