ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

A Review on Detection Mechanisms for SQL Injection Attacks

Aanal Bhanderi [1], Nency Rawal [2]

P.G. Student, Dept. of Computer Engineering, C.G.P.I.T, Uka Tarasadia University, Bardoli, Gujarat, India[1]

Associate Professor, Dept. of Computer Engineering, C.G.P.I.T, Uka Tarasadia University, Bardoli, Gujarat, India[2]

ABSTRACT: SQL Injection Attacks (SQLIAs) are one of the most serious security threats of database driven applications. SQL Injection was rated as the top most attack on the OWASP (Open Web Application Security Project). In SQL injection the attacker inserts structured query language code to input of a web form to gain access to database. SQL injection vulnerability allows an attacker to access commands to a web application's database and destroy confidentiality. Researchers have published number of tools for SQLIA detection. Some of which are AMNESIA, MySQLInjector, Paros 3.2.12 etc. In this paper, we present a detailed overview on various types of SQL injection attacks, vulnerabilities, and detection techniques. A comparative analysis of these techniques is based on some parameters to detect the SQL Injection attack is shown in paper.

KEYWORDS: SQL injection attacks, Web application, Detection techniques, Tools.

I. INTRODUCTION

In today's world use of internet is increasing day by day like many people do online transaction and online banking. Along with growing use of internet, there are many possibilities of attacks on web applications. SQL Injection Attacks (SQLIAs) are most common threat of web application. According to Open Web Application Security Project (OWASP) [10] top 10 threats for web application security in 2013 SQLIAs are top most.

SQL Injection attack exploits security vulnerability. Attacks occur in the database layer of an application [6]. It also allows attacker to leak sensitive information. Detection of SQL injection attacks is required to stop this type of harmful attack. To achieve this objective, automatic tools have been implemented by different authors which will be discuss in related work. The structure of this paper is as follows.

Chapter 1 describes definition and brief introduction of SQL Injection attack. Related work of SQL Injection detection tools are given in Chapter 2. Chapter 3 provides comparative analysis between SQL Injection detection tools. Chapter 4 finally summarizes a conclusion and future perspectives of this survey.

1. What is SQL Injection attack?

SQL Injection is a technique in which attacker injects an input query in order to change the query and illegally gain the access of the database [12]. SQL Injection allows attackers to create, read, update, alter, or delete and modify query in the back-end database and it also allow attackers to access sensitive information such as social security numbers, credit card number and other financial data [7]. When any vulnerability present in web applications then the error is generated. Attacker takes an advantage of this error message as it displayed by the web server depicts the type of database structure that has been used [8].

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

2. Types of Vulnerabilities

There are different types of vulnerabilities in web applications hence SQL Injection attacks can possible [5]. Table 1 shows the different types of vulnerabilities. Attackers usually try to attack on web application using these vulnerabilities.

Table 1. Types of vulnerabilities [6]

Vulnerability Types	Description
Type I	Lack of clear distinction between data types accepted as input.
Type II	Delay of operation in the runtime phase with current variables is considered rather than the source code.
Type III	Weak concern of type specification in the design.
Type IV	The validation of the user input is not well defined.

3. Types of SQL Injection Attacks (SQLIAs)

Fig. 1 shows the most commonly known SQL Injection attacks. The brief descriptions of attacks are as follows.

1) Tautology [1]:
   - The main goal of tautology based attack is to inject code in conditional statements so that they are always evaluated as true. For example: SELECT * FROM user WHERE id='1' or '1=1'-'AND password='1234';"or 1=1"
   - In this example code injected at the WHERE clause and retrieve results because the WHERE clause is always true. If attackers succeed in this then some records of database will returned.

2) Blind SQL Injection [11]:
   - Sometimes developers hide the error details which help attackers to compromise the database. For example: SELECT accounts FROM users WHERE id= '1111' and 1 =0 – AND pass = AND pin=0 SELECT accounts FROM users WHERE login= 'doe' and 1 = 1 -- AND pass = AND pin=0.
   - In this example, if application is secured then both queries will unsuccessful because input validations are strong but if there is less or no input validations then attacker try to modify the query.

3) Incorrect queries [6]:
   - It generates error message as query is rejected and attacker use that error message for attack.
   - For example: Original URL: http://www.toolsmarketal.com/veglat/?id_nav=2234
   - In this example attacker makes a type mismatch error by injecting the following text into the input field.
   - SQL Injection: http://www.toolsmarket-al/veglat/? id_nav=2234'

4) Piggy backed queries [3]:
   - This attack is one of the harmful attacks. In this attack, attacker attack using delimiter so main query join with extra query. Piggy backed query example is given below.
   - Backend Process: Select * from table where user id=„some" and pass=„";drop table users -- ;
   - Database treats this query as two different queries are separated using the delimiter (';') and it executes both query. In this, the first query is original query and the second query is an injected query.

5) Alternative Encoding [4]:
   - There are various types of attacks use the special characters as input to the web application.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

- Alternative encoding allows attacker to modify their injected strings in such a way that they can avoid typical signature based and filter based checks so attacker makes use of encoding such as ASCII, hexadecimal and Unicode.

6) Union query [4]:

- In this attack, attacker join safe query with malicious query using union operation and then can get data about other tables from the application then it can retrieve data from database. In this query, the injected query will returned balance from employee table, whereas original query return the null set [4].
- "SELECT name FROM bank WHERE userid=' ' UNION SELECT balance FROM employee WHERE empid='123'AND pswrd=' ' AND pin= "

7) Stored procedure [4]:

- Stored procedure is the one part of database so developer can set extra abstraction layer in the database. The main goal of this attack is tries to execute stored procedures present in the database.

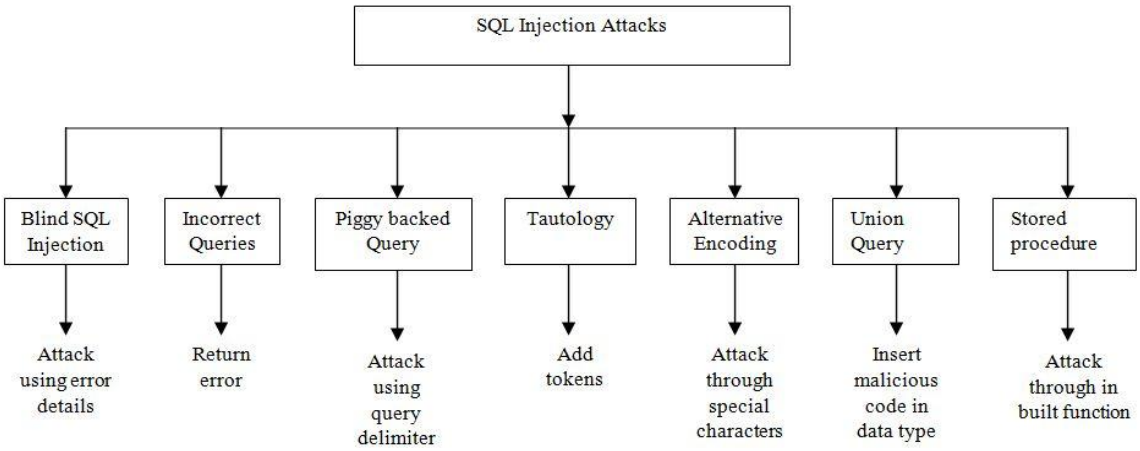

Fig. 1. Types of SQL Injection Attacks [13]

II. RELATED WORK

To protect the web applications from SQL Injection attacks, there are two major concerns. First, detection mechanisms are required to detect and identify SQL Injection attacks. Second, brief knowledge of SQL injection vulnerabilities are also require to protecting web application. In this section, we describe most relevant detection techniques and tools such as Analysis and Monitoring for NEutralizing SQL Injection Attack (AMNESIA), SQL injection vulnerability scanning tool for automatic creation of SQL Injection Attacks, a novel method for SQL injection attack detection based on removing SQL query attribute values [4]. Brief description of the detection techniques and tools are discussed.

A. Analysis and Monitoring for NEutralizing SQL Injection Attacks (AMNESIA)

AMNESIA technique is proposed by Halfond et. al. in [1], author suggested that AMNESIA is the effective SQLIAs detection tool. They also describe that it is model base technique which include both static analysis and dynamic analysis. AMNESIA technique combines static and dynamic phase for detecting web application vulnerabilities at the runtime. Static phase is used to generate different type of query statements. Dynamic phase is used to interpret all queries before they are sent to the database and validates each query against the statically built query models. AMNESIA stops all queries before they are sent to the database and validates each query statement against the AMNESIA models. Queries that violate the model represent potential SQLIAs and are thus prevented from executing on the database and reported. Fig. 2 shows detailed overview of AMNESIA.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

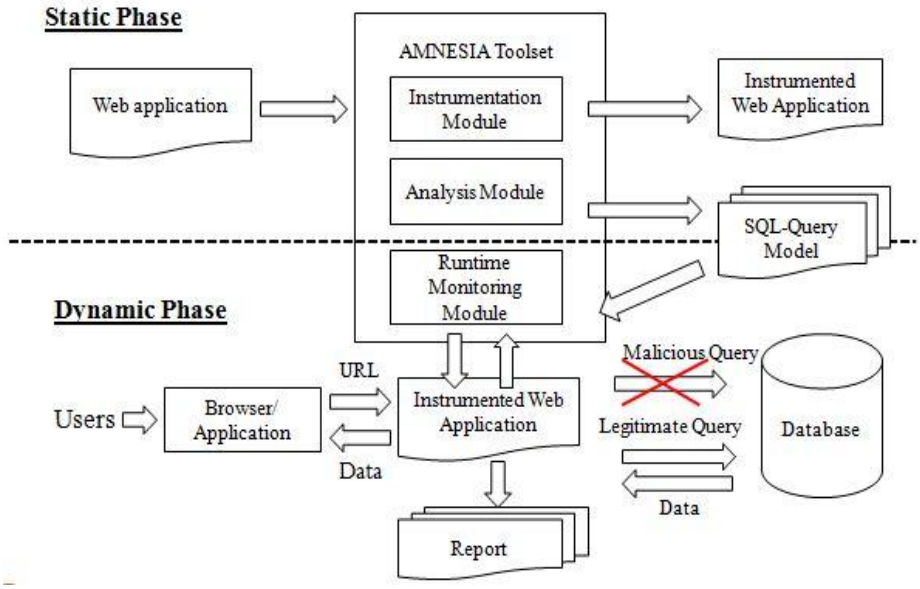

Fig. 2. Detailed flow of AMNESIA [1]

This technique consists of four main steps. Fig. 3 shows basic flow of AMNESIA technique.

1) Identify hotspots:
   - Application code is scan to identify hotspots points in the code which issue different SQL queries for underlying database.
2) Build SQL-query models:
   - After finding hotspot point, query model is build to represent all possible SQL queries that may be generated at that hotspot. SQL query model is a non-deterministic finite automation in which the transition labels consist of SQL tokens like SQL keywords and operators, delimiters.
3) Instrument Application:
   - In this step author Instrument the web application by adding a call to monitor, before the actual call to database is made. Web application is monitor using two parameters which is string and unique identifier and checks the query against the correct model.
4) Runtime monitoring:
   - At runtime, the application executes easily after identify hotspot point. Moreover it checks the dynamically generated queries against SQL query model and reject and report queries which are against to the query model.

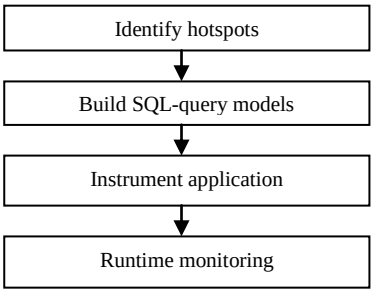

Fig. 3. Basic flow of AMNESIA [1]

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

B. SQL-injection vulnerability scanning tool for automatic creation of SQL Injection Attacks

MySQLInjector tool is described by Ali et.al. in [2]. They have mentioned that this tool is PHP based penetration test tool. Fig. 4 shows detection steps of MySQLInjector tool which describe as follows.

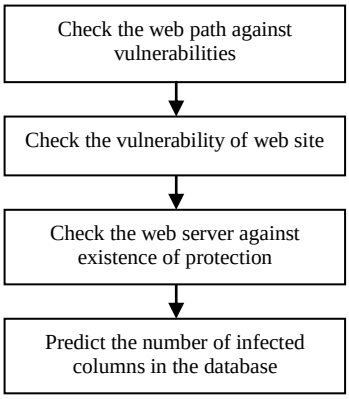

Fig. 4. MySQLInjector tool [2]

MySQLInjector tool is used to conduct penetration test on PHP based websites to detect hidden SQL Injection vulnerabilities.

- Penetration test is conduct as to ensure that single entry point to web application consider as week attack on websites but more than one entry points means chances of hacking is greater.
- Attacking patterns are important to expose vulnerabilities. It can detect hidden SQL vulnerabilities using three motioned features in multiple levels with high sensitivity parser.
- It is client side tool so it cannot conduct additional tool.

C. A novel method for SQL injection attack detection based on removing SQL query attribute values

This technique is proposed by Lee et. al. in [4]. SQL Injection detect using Paros 3.2.10 tool is described. Author propose novel method to detect SQL Injection attack, it removes the value of SQL query attributes of web pages when parameters compare with predefined query. This method works as follows:

- Removes the value of an SQL query attribute of web pages when parameters are submitted at runtime.
- Compares SQL query attribute with the SQL queries analysed in advance.
- At the end, detecting SQL injection attacks by comparing static SQL queries with dynamically generated queries after removing the attribute values
- This method cannot only be implemented on web applications but it can also be used on any applications connected to databases.
- It can be used for SQL query profiling, SQL query listing and modularization of detection programs.

Author proposes useful detection approach using combination of both static and dynamic analysis. In static analysis, it analyses the SQL query sentences of web applications to detect and prevent web applications from SQL Injection attack. In dynamic analysis, it analyses the response of web applications after scanning it can locate vulnerabilities from SQLIAs without making modifications to web applications [4]. Attacks are detected by comparing grammar from queries.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

For example, if dynamically generated query has different structure from static query then it is detected. This method checks SQL queries which attribute values have either string values or numeric values. Paros 3.2.12 tool is used as an attack detection rate tool [4]. Detection frequency v/s attack frequency is defined as detection rate. Advantage of this method is it works as automated detection and prevention of SQL Injection attack. Disadvantage of this method is it cannot detect SQL Injection attacks in real time.

III. COMPARATIVE ANALYSIS

Based on literature survey detection techniques can mostly detect SQLIAs at runtime. AMENESIA is static and dynamic detection tool which detects tautologies, incorrect queries, piggy backed query, alternative encoding, union queries and it is a server side application that supports JSA. MYSQL Injector is dynamic detection tool which detects blind SQL injection attack and it is client side technique that supports PHP. Paros 3.2.13 is static and dynamic detection tool which detects tautologies, incorrect queries, piggy backed query, alternative encoding, union queries and it is a server side application.

From that we conclude that in first technique AMNESIA, author works on both static and dynamic analysis for detect critical SQL Injection attacks. It also provides detection as well as defence mechanisms. This technique provides false positive and false negative rate and SQL query model is helpful to detect malicious queries. In second paper which is based on MySQLInjector tool, author use attacking patterns to detect hidden SQL vulnerabilities but it is only base on PHP base applications so in future it can implement on other web applications. In third paper which is based on Paros 3.2.12 tool, author compare statically generated queries with dynamically generated queries so by comparison malicious query can rejected but it cannot detect real time applications.

Table 2. Comparative analysis of SQLIAs detection tools

Tools	Type of detected Attacks	Detection	Detection Location
AMNESIA[1]	Tautology, incorrect queries, piggy back, alternative encoding, union query	Static and dynamic	Server side application
MySQL Injector[2]	Blind SQL Injection attack	Dynamic	Client side application
Paros 3.2.13[4]	Tautology, incorrect queries, piggy back, alternative encoding, union query	Static and dynamic	Server side application

IV. CONCLUSION AND FUTURE SCOPE

SQL Injection is a common technique that hacker employ to attack on web based applications. These attacks modify the SQL queries, so the behaviour of the program is altering which provide the benefit of the hacker. This paper presents various types of SQLIA and explores SQL injection detection tools in related work. Comparison is carried on these tools in terms of their ability to detect the SQLIA. Furthermore, the tools were compared based on its detection parameters. These detection techniques are helped to detect this type of harmful attack and AMNESIA is best for SQLIAs detection as it can be detect all types of attacks at both static and dynamic phase.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 12, December 2015

In future, these detection tools can be used to detect SQL Injection attacks. These techniques can also provide as defence mechanisms for provide security against SQLIAs. In addition, more research is needed to improve analysis technique for providing precise detection and defence against strong SQLIAs.

REFERENCES

[1] Halfond, GJ, Orso. "AMNESIA: analysis and monitoring for NEutralizing SQL-injection attacks." In Proceedings of the 20th IEEE/ACM international Conference on Automated software engineering, pp. 174-183. ACM, 2005

[2] Ali, Mat, Abdullah, Alostad. "SQL-injection vulnerability scanning tool for automatic creation of SQL-injection attacks." Procedia Computer Science 3,pp. 453-458, Elsevier-2011

[3] Gupta, M. K., Govil, Singh. "Static analysis approaches to detect SQL injection and cross site scripting vulnerabilities in web applications: A survey." In Recent Advances and Innovations in Engineering (ICRAIE), pp. 1-5. IEEE, 2014.

[4] Lee, Inyong, Soonki Jeong, Sangsoo Yeo, and Jongsub Moon. "A novel method for SQL injection attack detection based on removing SQL query attribute values." Mathematical and Computer Modelling , no. 1, pp. 58-68, Springer (2014).

[5] Singh, Garima, Kant, Gangwar, and Singh. "SQL Injection Detection and Correction Using Machine Learning Techniques." InEmerging ICT for Bridging the Future-Proceedings of the 49th Annual Convention of the Computer Society of India (CSI) Volume 1, pp. 435-442. Springer International Publishing, 2015.

[6] Gupta, M. K., Govil, and Singh. "Static analysis approaches to detect SQL injection and cross site scripting vulnerabilities in web applications: A survey." In Recent Advances and Innovations in Engineering (ICRAIE), pp. 1-5. IEEE, 2014.

[7] Li, Xiaowei, and Yuan Xue. "A survey on server-side approaches to securing web applications." ACM Computing Surveys (CSUR), no. 4 (2014).

[8] Dwivedi, Vandana, Himanshu Yadav, and Anurag Jain. "Web Application Vulnerabilities: A Survey." International Journal of Computer Applications , (2014).

[9] Dan Bonesh. SQL Injection attacks and defense [online]. Available at: https://crypto.stanford.edu/cs142/lectures/16-sql-inj.pd

[10] Mark Churphy. The Open Web Application Security Project, "OWASP TOP Project" [online]. Available at: https://www.owasp.org/SQL_Injection

[11] Rafique, Sajjad, Mamoona Humayun, Bushra Hamid, Ansar Abbas, Muhammad Akhtar, and Kamil Iqbal. "Web application security vulnerabilities detection approaches: A systematic mapping study." In Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 16th IEEE/ACIS International Conference on, pp. 1-6. IEEE, 2015.

[12] Razzaq, Abdul, Khalid Latif, H. Farooq Ahmad, Ali Hur, Zahid Anwar, and Peter Charles Bloodsworth. "Semantic security against web application attacks."Information Sciences 254, pp. 19-38 (2014).

[13] Kindy, Abdoulaye, and Pathan. "A survey on SQL injection: Vulnerabilities, attacks, and prevention techniques." (2011).