Int'l Conf. on Science, Engineering and the Social Science (ICSESS 2015)
11 – 13 May, 2015, UTM Johor Bahru, Malaysia

SC03

CRITICAL ISSUE TO CONSIDER WHILE DEVELOPING SQL INJECTION PREVENTION MECHANISM

Muhammad Saidu Aliero *, Dr. Imran Ghani , Muhammad Murad Khan, Nkima L.H
Universiti Teknologi Malaysia.
msaidua2000@gmail.com

Keywords SQL Injection, Injection. Parameter , Defensive Coding

Abstract: SQL injection vulnerability is the one of the most common web-based application vulnerabilities that can be exploited by SQL injection attack to gain access to restricted data, bypass authentication mechanism and execute unauthorized data manipulation language. Defensive coding is the simple and affordable way to tackle this problem, by applying secure coding in each an every queries used in application. In this paper we provide a detailed background of SQLI attack, we classify defensive coding into different categories, review existing techniques that are related to each technique, and also evaluate such techniques based on number of attacks they were able to stop.We also evaluated each category of approach based on it's deployment requirement related to inheritance. Currently, to the best of our knolwdge no papers have classied defensive coding as we do.

I. INTRODUCTION

SQLI (SQL injection) vulnerability is the one of the most dangererous vulnerablities in web-based database driving applications. It occurs as a result of inappropriate user input validation, which enables the attacker to manipulate programmer intended queries by adding new SQL operator, command, keyword, or clause to perform unauthorized database extraction modification, thereby bypassing authentication mechanism.

The main cause of SQL injection vulnerability is improper validation of user input. Input validation is a technique by which a programmer applies defense code practice to secure each static query manually. One of the objectives of defensive programming is to write secure queries so that it behaves in a predictable manner despite unexpected inputs or user actions. It is based on the idea that every program module is solely responsible for itself. After web development, its code should be reviewed by security analysts for proper use of function.

II. BACKGROUD OF SQLIA

Injection Parameter

Injection through User input field: user input fields are provided in web applications to enable web application users to request information that are stored in application database to the user with help of HTTP POST and GET such as Login input, URL input and SEARCH input.

SQLIA Types

1. Tautology attack
2. Piggy-backend query attack
3. Illegal/Incorrect logical query attack
4. Inference attack
5. Store procedure attack
6. Union query attack
7. Alternate encoding attack

III. CATEGORAZATION OF DEFENSIVE CODING

Basically defensive coding can be characterized as shown in Figure 1 below. Programmer uses one or more approaches to patch up their application depending on the need of the enterprise. For example some enterprises do not allow search box in the website to reduce the risk of being attacked. In such cases they provide the user with predefined search inputs, thus there is no need of applying security code in search box. As shown not all approaches are applicable in all injection points.

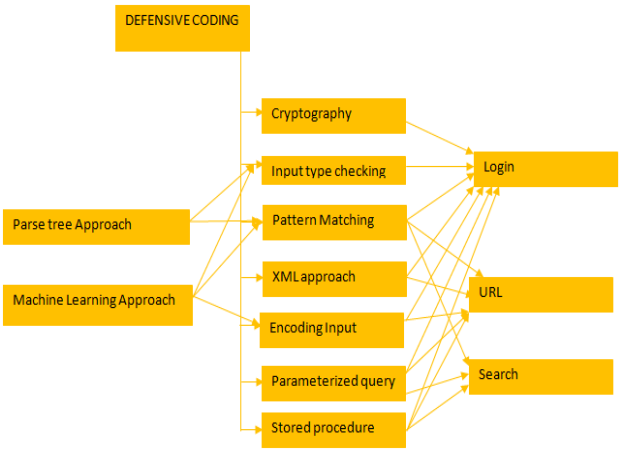

Figure 1 Categorization of Defensive Coding.

IV. EVALUATION

Evaluation Based on Deployment

We analyze each approach as shown in Figure 2 based on different deployment requirements.

METHOD	URL	LOGIN	SEARCH	DETECT	PREVENT	MODIFY CODE BASE	RESISTANCE TO ATTACK	ACCURACY
CRYPTOGRAPHY	X	□	X	X	□	□	HIGH	HIGH
PATTERN MATCHING	□	□	□	□	□	X	MEDIUM	DEPEND ON FILTER PATTERN
XML	□	□	□	X	□	X	LOW	HIGH
MASHINE LEARNING	□	□	□	□	□	□	MEDIUM	DEPEND ON TRAINING DATA
PARSING	□	□	□	□	□	□	HIGH	DEPEND ON TREE STRUCTURE

Figure 2 Evaluation based on deployment requirments.
"◦"indicate method can be deployed to that injection parameter.
"x"indicate method cannot be deployed to that injection parameter.

Evaluation Based on the Attack Type

We analyzed and evaluate each proposed method as shown in Figure 3 to assess whether it is capable of addressing particular attack. Evaluation was done analytically based on our experience,:we have not assess any of the method in real time practices because implementation codes of most methods are are not available or some methods are not implemented.

METHOD	tautology	Illegal/incorrect	Piggy-backend	inference	Alternate encode	Stored procedure	Union
LITERATURE[1]	□	□	X	□	X	X	X
LITERATURE [2]	□	□	X	□	X	X	X
LITERATURE 3]	□	□	X	□	X	X	X
LITERATURE [4]	□	□	X	X	X	X	X
LITERATURE [5]	□	□	□	x	□	x	□
LITERATURE [6]	□	□	□	x	□	x	□
LITERATURE [7]	□	□	□	x	□	x	□
LITERATURE [8]	□	□	□	x	□	□	□
LITERATURE [9]	□	□	X	□	X	□	X
LITERATURE [10]	□	□	□	□	□	X	□
LITERATURE [11]	□	□	□	X	□	X	□
LITERATURE [12]	□	□	□	X	□	X	□
LITERATURE [13]	□	□	□	X	□	□	□
LITERATURE [14]	□	□	□	X	□	□	□
LITERATURE [15]	□	□	□	□	□	□	□
LITERATURE [16]	□	□	□	X	□	X	□
LITERATURE [17]	□	□	□	X	□	X	□
LITERATURE [18]	□	□	□	x	□	□	□
LITERATURE [19]	□	□	□	x	□	x	□
LITERATURE [20]	□	□	□	X	□	X	□

Figure 3 Evaluation Based on the Attack Type

"◦"indicate method can successfully stop attack of that type.
"x"indicate method cannot stop attack of that type.

Conclution

In this paper we present background of SQLIA. We introduced most common approach used to prevent SQLIA, categorized such approaches into a survey of different methods, surveyed existing techniques related to each method, identify some common issues related to each approach as well as programmer mistakes. We also evaluated each approach based on its inherited deployment requirement and each technique based on the type of attack able to address.

REFERENCE

[1]Mihir Gandhi, JwalantBaria 2013. SQL INJECTION Attacks in Web Application *International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-6.*

[2] Raghav Kukreja , Nitin Garg 2014. OVERVIEW OF SQL INJECTION ATTACK *international journal of innovative research in technology Volume 1 Issue 5.*

[3] Ms. Mira K. Sadar et al 2014. Securing Web Application against SQL Injection Attack: a Review *International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169 Volume: 2 Issue: 3*

[4] Neha Mishra , Sunita Gond 2013. Defenses To Protect Against SQL Injection Attacks International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 10

[5] R. Joseph Manoj et al 2014. An Approach to Detect and Prevent Tautology Type SQL Injection in Web Service Based on XSchema validation. *International Journal Of Engineering And Computer Science ISSN:2319-7242 Volume 3 Issue 1*

[6]Shrivastava, Rahul, Joy Bhattacharyji, and RoopaliSoni2013."SQL INJECTION ATTACKS IN DATABASE USING WEB SERVICE: DETECTION AND PREVENTION–REVIEW." *Asian Journal of Computer Science & Information Technology* 2.6

[7]IndraniBalasundaram, E. Ramaraj 2011. "An Approach to Detect and Prevent SQL Injection Attacks in Database Using Web Service." *IJCSNS International Journal of Computer Science and Network Security* 11.1 95-100.

[8]Borade, Monali R., and Neeta A. Deshpande2014. "Web Services Based SQL Injection Detection and Prevention System for Web Applications." *International Journal of Emerging Technology and Advanced Engineering (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue*

[9]Indrani, B., and E. Ramaraj.(2011) "X–LOG AUTHENTICATION TECHNIQUE TO PREVENT SQL INJECTION ATTACKS." *International Journal of Information Technology and Knowledge Management* 4.1 : 323-328.

[10]Das, Debasish, Utpal Sharma, and D. K. Bhattacharyya (2010) "An Approach to Detection of SQL Injection Attack Based on Dynamic Query Matching."*International Journal of Computer Applications* 28-34.

[11]Prabakar, M. Amutha, M. Karthikeyan, and K. Marimuthu 2013. "An efficient technique for preventing SQL injection attack using pattern matching algorithm." *Emerging Trends in Computing, Communication and Nanotechnology (ICE-CCN), 2013 International Conference on*. IEEE,

[12]Narayanan, Sandeep Nair, AlwynRoshanPais, and Radhesh Mohandas 2011."Detection and Prevention of SQL Injection Attacks Using Semantic Equivalence." *Computer Networks and Intelligent Computing*. Springer Berlin Heidelberg, 2011. 103-112.

[13]Ms. Zeinab Raveshi and Mrs. Sonali R. Idate 2013 Efficient Method to Secure Web applications and Databases against SQL Injection Attacks

[14]Manmadhan, Sruthy, and T. Manesh(2012) "A method of detecting sql injection attack to secure web applications." *International Journal of Distributed and Parallel Systems* 3 1-8.

[15]Kumar, Kuldeep, Debasish Jena, and Ravi Kumar(2013). "A Novel Approach to detect SQL injection in web applications." International Journal of Application or Innovation in Engineering & Management (IJAIEM) Volume 2, Issue ISSN 2319 - 4847

[16]Bangre, Shruti, and AlkaJaiswal(2012) "SQL Injection Detection and Prevention Using Input Filter Technique." *International Journal of Recent Technology and Engineering (IJRTE)*145-149.

[17]Shi, Cong-cong, et al.(2012) "A New Approach for SQL-Injection Detection."*Instrumentation, Measurement, Circuits and Systems*. Springer Berlin Heidelberg,. 245-254.