Scientific Journal of Impact Factor (SJIF): 3.134

ISSN (Online): 2348-4470
ISSN (Print) : 2348-6406

International Journal of Advance Engineering and Research Development

Volume 2, Issue 1, January -2015

A Review on Approaches for Web Application Vulnerabilities Detection

Antra Mahadkar[#1], Mr. Narendra Singh*[2]

[#*]*Computer Science & Engineering,* [#*]*Parul Institute of Engineering and Technology*
[#*]*Gujarat Technological University*
[1]*annumahadkar@gmail.com* ,[2]nsingh9011@gmail.com

Abstract *—Web application i used for sharing information and web services over internet today. The popularity of web applications attracts attackers to attack on them and attackers are well known about the vital information which accessible with the help of web application which is very serious security attacks on web applications.*
In this paper, we survey the different approaches for detection of vulnerabilities in web application and services. At the last we summarize the approaches and discussed the future scope

KEY WORDS: *Web Apllication, Web Services, Vulnerability, Attack*

I. INTRODUCTION

The most future attacks will be at the application level, where security mechanisms are inadequate. Web application provides interaction between client and server through the web pages which contains the HTML, images, code, etc. for users' attraction but also exploits the security vulnerabilities. Vulnerability is a cyber-security term that refers to a flaw in a system that can leave it open to attack. A vulnerability may also refer to any type of weakness in a computer system, in a set of procedures or in anything that leaves information security exposed to a threat. Therefore It is necessary to protect web application from vulnerabilities. The security of the web application is very important.

SQL injection, Cross-site scripting(XSS), Password authentication mechanism, Denial of service vulnerability, Insecure masked password storage vulnerability, Information disclosure vulnerability, Broken access control, Session management, Unvalidated redirects and forwards, security misconfiguration, etc. Some of these are present in OWASP Top 10 vulnerabilities. The OWASP Top 10 focuses on the identifying the most serious vulnerabilities for the broad array of the organizations [8].

II. BACKGROUND

A. Vulnerabilities

Vulnerability is a flaw or weakness in a system's design, implementation, operation or management that could be exploited to compromise the system's security objectives. A vulnerability may also refer to any type of weakness in a computer system, in a set of procedures or in anything that leaves information security exposed to a threat. Therefore It is necessary to protect web application from vulnerabilities. The security of the web application is very important.

The Vulnerability Checking approaches are as follows:

Error Pattern Matching**:** To identify the vulnerability in web application, this approach looking for specific patterns in the responses: database error messages returned by web server/database server while the specially crafted requests sent to the application. The basic idea is that the presence of an SQL error message in a HTML response page means that the corresponding request has not been sanitized by the application. Therefore, the fact that this request has been sent unchanged to the SQL server reveals the presence of vulnerability.

Similarity Approach**:** To identify the vulnerability in web application, this approach compares the similarity of the corresponding responses from the server while the specially crafted requests sent to the application. The comparison uses a textual distance in order to identify the execution pages among the response pages.

Cluster Approach**:** To differentiate rejection page and execution page this approach is used. It is based on the automatic classification of the responses returned by the web servers using data clustering techniques and provides especially crafted inputs that lead to successful attacks when vulnerabilities are present.

Parse Tree Validation: This is proposed approach in this project, which will take all possible valid queries generate from the valid query model and Validation/Malicious Queries generated from the Validation query model as input, will be parsed in the tree structure. By parsing two queries and validating their tree structures and using some Attack vector's/signature's regular expression, it can be determined if the two queries are intend to same meaning and also the respective input field is vulnerable or not.

B. SQL/XPath Injection detection using different approaches

1) Effective Detection of SQL/XPath Injection Vulnerabilities in Web Services[2]

They proposes a new approach for detection of most critical types of vulnerabilities : SQL injection and XPath injection. A representative workload is used to exercise the web service and a large set of SQL/XPath Injection attacks are applied to disclose vulnerabilities. Vulnerabilities are detected by with the comparison of structure of the SQL/XPath commands which were issued in the presence of attacks to the ones previously learned when running the workload in the absence of attacks. Experimental evaluation shows that their approach performs much better than known tools (including commercial ones), achieving extremely high detection coverage while maintaining the false positives rate very low. But they uses Pattern matching process. In this process hash code of the each and every query will bw generated and then it compare with the learned hash code command set. If match is found then that is being executed, otherwise the occurrences is logged for the future reference.
Disadvantage of this process is comparison time is very high, therefore response time increases for the client. But they gives good false negative and coverage.

2) Using Web Security Scanners to Detect Vulnerabilities in Web Services[3]

In this paper , they uses web security scanners for detection of vulnerabilities. Total four scanners are used , VS1.1 , VS1.2 , VS2 and VS3. VS1.1 and VS1.2 are versions of the V1 scanner. Scanners have been used to detect security flaws in web services and applications implementations too.

They did analysis between false positive rate and coverage using vulnerability scanners.
FP means query is not vulnerable to web application but scanner detects it as vulnerability. And coverage means, how much vulnerability are detected agains total number of vulnerabilities. They achieved low coverage and high FP rate. So this is not effective for detection of vulnerabilities. Here VS1.1 detects SQL injection vulnerabilities but VS2 and VS3 cannot detect SQL/XPath injection vulnerabilities. Some of the vulnerabilities are detected by VS1.2 but not by VS1.1.

Table 1.1 Result generated using Scanners

Sr. No.	Scanners	Vulnerabilities	
		SQL Injection	XPath Injection
1	VS1.1	✓	✗
2	VS1.2	✗	✓
3	VS2	✗	✗
4	VS3	✗	✗

Additionally, the differences in the vulnerabilities detected and the high number of false-positives and low coverage observed highlight the limitations of web vulnerability scanners on detecting security vulnerabilities in web services.

3) A clustering approach for web vulnerabilities detection [4]

This paper presents a new algorithm aimed at the vulnerability assessment of web applications following a blackbox approach. Their approach covers various types of vulnerabilities but this paper mainly focuses on SQL injections. The proposed algorithm is based on the automatic classification of the responses returned by the web servers using data clustering techniques and provides especially crafted inputs that lead to successful attacks when vulnerabilities are present.
Sever respond with a rejection page or with an execution page. Here, if number of requests increases, then it goes in rejection pages.

4) A Learning-Based Approach to Secure Web Services from SQL/XPath Injection Attacks[5]

This paper proposes a learning based approach to secure web services against SQL injection and XPath injection attacks. Their approach is able to learn valid request patterns i.e. learning phase and then detect and eliminate harmful requests i.e. in peotection phase.

They include two major phases.

- Statement Learning
- Service Protection

5) Using Parse Tree Validation to Prevent SQL Injection Attack[6]

They proposed a technique to avoid SQL injection attack by comparing at runtime, the parse tree of the SQL statement before inclusion of the user input with the resulting after inclusion of the input.

They proposes approach ,which will take all possible valid queries generate from the valid query model and Validation/Malicious Queries generated from the Validation query model as input, will be parsed in the tree structure. By parsing two queries and validating their tree structures and using some Attack vector‟ s/signature‟ s regular expression, it can be determined if the two queries are intend to same meaning and also the respective input field is vulnerable or not.

6) Preventing SQL Injection Attacks Using AMNESIA[7]

AMNESIA (Analysis and Monitoring for Neutralizing SQL injection Attacks), is a tool that uses a model based approach designing to detect SQL injection attacks and combines static analysis and runtime monitoring. Static analysis is used to build a model of legitimate queries that an application can generate. At runtime when a query that violates the model is detected, it is classified an attack and is prevented from accessing the database. The proposed approach will take all the possible valid queries as input and will learns the structure of valid queries by parsing them as parse tree and then it will validate the clients/validation queries at run time, overcoming the intrinsic limitations of static analysis.

III. CONCLUSION

From the survey of various papers it is found that SQL Injection and XPath Injection are most powerful attack methods on the web application. This survey presents the current approaches for detecting against SQL Injection and XPath exploitation. Our review finds that existing approaches suffer from one or more of the following:

- False positives and False negatives
- Low coverage
- Time consuming
- Complex

All existing approaches are effective but not efficient.

REFERENCES

[1] https://www.owasp.org/index.php/Category:Vulnerability

[2] Antunes N., Laranjeiro N., Vieria M. and Maderia M. (2009). "Effctive Detectionn of SQL/XPath Injection Vulnerabilities in Web Services", IEEE International Conference on Services Computing.

[3] Vieira,M., Antunes,N.and Madeira,H.(2009) "Using web security scanners to detect vulnerabilities in web services" Dependable Systems & Networks, DSN '09. IEEE/IFIP International Conference

[4] Dessiatnikoff, A. ; Akrout, R. ; Alata, E. ; Kaaniche, M. ; Nicomette, "A Clustering Approach for Web Vulnerabilities Detection"- 2011 Dependable Computing (PRDC), 2011 IEEE 17th Pacific Rim International Symposium

[5] William G., J. Halfond and A. Orso "Preventing SQL Injection Attacks Using AMNESIA":PublicationYear:2006

[6] Nuno Laranjeiro, Marco Vieira, Henrique Madeira "A Learning-Based Approach to Secure Web Services from SQL/XPath Injection Attacks" PublicationYear:IEEE-2011

[7] Gregory T. Buehrer, Bruce W. Weide, and Paolo A. G. Sivilotti, "Using Parse Tree Validation to Prevent SQL Injection Attacks" , Computer science & Engineering Dept., The Ohio State University

[8] https://www.owasp.org/index.php/Top_10_2013-Top