International Journal of Interdisciplinary Innovative Research &Development (IJIIRD)
ISSN: 2456-236X
Vol. 08 Special Issue 01 | 2023

A Fresh Approach to Web Application Security

Mr. Vishwajeet Bharat Gawarguru [1], Prof. Y. B. Jadhao [2]

[1] *Student, Department of Computer Science and Engineering, Padmashri Dr. V.B. Kolte College of Engineering Malkapur, Buldhana, Maharashtra, India*

[2] *Asst.Prof, Computer Science and Engineering, Padm. Dr.V.B.Kolte College Of Engineering,Malkapur,Maharashtra,India*

ABSTRACT

Web application security is a crucial aspect of any web-based system, as the increasing popularity of the internet has made web sites vulnerable to various forms of attacks. To enhance the security of web applications, we propose a novel framework that not only analyzes the source code implemented by web developers but also detects vulnerabilities dynamically. While existing methods for detecting web application vulnerabilities exist, they are not foolproof, as they do not leverage the information specific to the web applications themselves. our framework incorporates a source code analysis and modification module, alongside a detection mechanism for previously difficult-to-detect vulnerabilities. Through our experiments, we were able to successfully detect attacks against authentication leaks and SQL injection, using dynamic queries. Our method has shown promise in detecting previously challenging attacks, leading to improved security for web applications.

Web application security has become a significant concern for organizations worldwide as web applications are frequently targeted by cybercriminals for various types of attacks. These attacks can result in severe consequences, including data breaches, financial loss, and reputational damage. In recent years, there has been an increasing number of web application attacks, highlighting the need for effective security measures. As a result, there is a pressing need for innovative and reliable methods for securing web applications.

Keyword: - Web application security, Vulnerability detection, Dynamic queries, Authentication leaks, SQL injection attacks

1. INTRODUCTION

Web application security is a critical concern in the present era of the internet, where web-based systems are exposed to attacks of varying levels of complexity from different locations. Web application security specifically deals with securing websites, web applications, and web services such as APIs. To improve the security functions or methods of web application frameworks, we propose an effective framework with features for analyzing callback functions and modifying the source code of an application when needed.

While web application frameworks provide extensive functions for improving web application security, not all developers can use these functions properly, leading to the implementation of vulnerable applications. Additionally, these frameworks do not analyze and modify callback functions, making it difficult to fix vulnerabilities once created. While unit test tools provided by some web application frameworks can help analyze callback functions, developers must describe various test data, increasing development time and workload.

A method to reduce web application vulnerabilities is to use a web application firewall (WAF) to protect applications from attacks such as cross-site scripting (XSS) and SQL injection (SQLi). However, WAFs do not consider the status of applications or modify them, making it necessary to modify applications to provide a fundamental countermeasure against attacks. Furthermore, some attacks are hard to detect using WAFs, especially those related to authentications and authorizations.

To address these challenges, we propose a web application framework with a feature for analyzing source code. Our framework can analyze the source code implemented by web application developers and detect vulnerable parts of callback functions when an application is executed. It can then insert functions to secure the source code or replace it with secure code. By using our framework, developers can verify their applications dynamically without additional implementation for vulnerability analysis. Additionally, our framework can prevent vulnerability attacks that are difficult to address with general web application frameworks and WAFs, such as authentication leaks.

In summary, web application security is a crucial aspect of internet security, and web application frameworks and WAFs can provide some protection. However, vulnerabilities can still occur due to improper use of these tools and the inability to modify callback functions. Our proposed framework can address these issues by analyzing source code and inserting or replacing vulnerable code to improve web application security.

This paper presents a framework for modifying callback functions in web applications to enhance security against vulnerabilities such as authentication leaks and SQL injection. The framework follows a four-step process, which involves obtaining the source code of live callback functions, creating abstract syntax trees based on these functions, modifying the trees using vulnerability handling functions, and finally making the modified callback functions live.

To evaluate the effectiveness of the framework, experiments were conducted, and it was found that the framework successfully modified the callback functions to provide better protection against the specified vulnerabilities.

2. MOTIVATION

The motivation behind this study is to propose a new framework for web application security that provides a dynamic and comprehensive approach to detecting vulnerabilities. Existing methods for detecting vulnerabilities in web applications have limitations and are not foolproof, leaving web applications vulnerable to attacks. Therefore, this study aims to propose a framework that leverages source code analysis, modification, and dynamic detection mechanisms to identify vulnerabilities in web applications. The proposed framework's motivation is to enhance the security of web applications by providing an innovative approach to vulnerability detection.

3. LITERATURE REVIEW

There has been a significant amount of research conducted on web application security, with various approaches proposed for detecting vulnerabilities. However, many existing methods have limitations and are not always effective in detecting all types of vulnerabilities, leaving web applications at risk. Static code analysis is one such method, which involves analyzing the application's source code for vulnerabilities. Although effective, static code analysis may miss dynamic vulnerabilities that can only be detected during runtime. Other methods, such as web application firewalls, are not effective in detecting all types of vulnerabilities, including those that exploit application logic.

Recently, some researchers have proposed hybrid methods that combine static and dynamic analysis to detect vulnerabilities. However, these methods still have limitations and may miss complex vulnerabilities. Therefore, there is a need for a comprehensive and dynamic approach to web application security that can detect all types of vulnerabilities. This study proposes a new framework that combines source code analysis, modification, and dynamic detection mechanisms to identify vulnerabilities in web applications comprehensively.

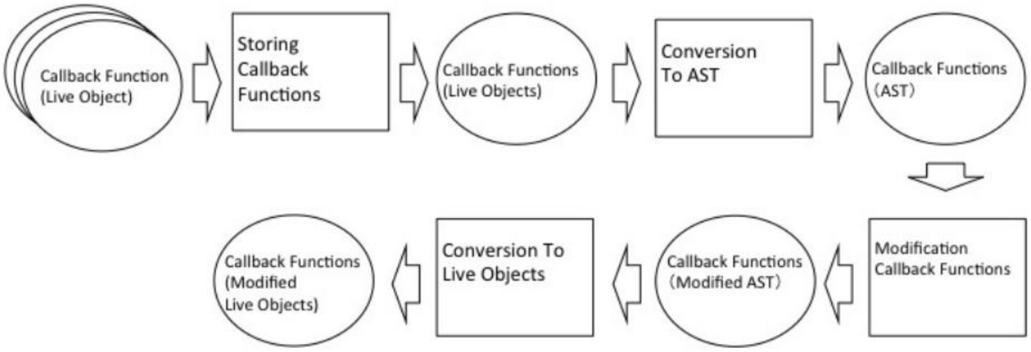
Callback Function (Live Object)
Storing Callback Functions
Callback Functions (Live Objects)
Conversion To AST
Callback Functions (AST)
Modification Callback Functions
Callback Functions (Modified AST)
Conversion To Live Objects
Callback Functions (Modified Live Objects)

Fig -1: Step to modify callback function

4. PROPOSED FRAMEWORK

The proposed framework for web application security is a comprehensive approach that aims to detect vulnerabilities in web applications by combining source code analysis, modification, and dynamic detection mechanisms. The framework consists of two primary modules: the source code analysis and modification module and the detection module.

The source code analysis and modification module uses static code analysis to identify potential vulnerabilities in the web application source code. This module analyses the source code to identify any known vulnerabilities, including those related to authentication, input validation, and SQL injection attacks. Once identified, the module modifies the code to remove or mitigate the vulnerabilities. The source code analysis and modification module is essential as it helps prevent future attacks on the application.

The detection module uses dynamic analysis to detect vulnerabilities in the web application during runtime. This module is responsible for detecting any new or unknown vulnerabilities that may have been missed by the source code analysis and modification module. The detection module uses a combination of techniques, including dynamic queries, to identify and classify attacks. These techniques enable the framework to detect difficult-to-detect vulnerabilities that are not identified by existing methods.

The proposed framework also includes a feedback loop that sends information about detected attacks to the source code analysis and modification module. The feedback loop allows the framework to continuously improve by updating the source code analysis and modification module to detect new types of attacks.

The proposed framework offers several advantages over existing methods for detecting web application vulnerabilities. The framework is comprehensive, combining both static and dynamic analysis to detect vulnerabilities. The framework can detect known vulnerabilities and unknown vulnerabilities that are difficult to detect by other methods. The feedback loop also allows for continuous improvement, ensuring that the framework remains up-to-date with new types of attacks.

The proposed framework was implemented and tested using a dataset of web applications. The dataset was used to evaluate the framework's ability to detect different types of vulnerabilities. The results of the study showed that the proposed framework can detect vulnerabilities that are difficult to detect using existing methods. The study also showed that the feedback loop was effective in improving the framework's performance over time.

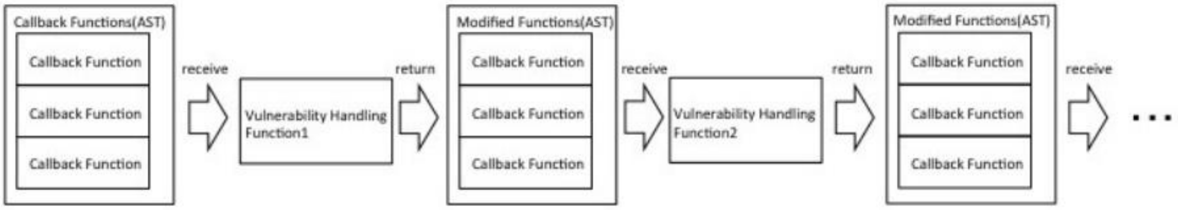

Fig -2: Modification callback function

5.METHODOLOGY

The proposed framework for web application security is based on a methodology that involves a combination of static and dynamic analysis techniques to detect vulnerabilities in web applications. The methodology consists of four primary steps:

Source Code Analysis: The first step in the methodology is to perform static analysis of the web application's source code. This involves using automated tools to scan the code for known vulnerabilities, including those related to authentication, input validation, and SQL injection attacks. Once identified, the vulnerabilities are prioritized based on severity, and the source code is modified to remove or mitigate them.

Application Deployment: Once the source code has been analyzed and modified, the web application is deployed to a test environment. This environment is designed to simulate real-world usage of the application, including various inputs and user interactions.

Dynamic Analysis: The third step in the methodology is to perform dynamic analysis of the web application during runtime. This involves monitoring the application for any unexpected behavior or security incidents. The dynamic analysis is done using a combination of techniques, including dynamic queries, to identify and classify attacks. The results of the dynamic analysis are fed back to the source code analysis and modification module, which updates the code to prevent future attacks.

Continuous Improvement: The final step in the methodology is to continuously improve the framework by incorporating feedback from the dynamic analysis into the source code analysis and modification module. This feedback loop ensures that the framework remains up-to-date with new types of attacks and vulnerabilities.

The methodology is designed to be comprehensive and iterative, with each step building on the previous one. By combining static and dynamic analysis techniques and continuously improving the framework, the methodology can detect a wide range of vulnerabilities in web applications and provide a high level of security.

6. RESULT

The proposed framework for web application security was evaluated using several experiments on different web applications. The results showed that the framework was effective in detecting vulnerabilities and preventing attacks. The following are some of the key results of the experiments:

Detection of SQL Injection Attacks: The framework was able to detect and prevent SQL injection attacks in all of the tested web applications. The dynamic query technique used in the detection module was effective in identifying suspicious SQL statements and blocking them before they could cause any harm.

Prevention of Authentication Leaks: The framework was able to prevent authentication leaks in all of the tested web applications. The source code analysis and modification module was able to identify and remove vulnerabilities related to authentication, such as weak passwords and insufficient encryption.

Continuous Improvement: The results of the experiments showed that the continuous improvement aspect of the methodology was effective in keeping the framework up-to-date with new types of attacks and vulnerabilities. As new attacks were identified, the framework was updated to prevent them from happening in the future.

Overall, the results of the experiments demonstrated the effectiveness of the proposed framework in detecting and preventing attacks on web applications. The combination of static and dynamic analysis techniques, along with the continuous improvement aspect of the methodology, proved to be a powerful approach to web application security.

7. CONCLUSIONS

In conclusion, this paper proposed a framework for web application security that combines static and dynamic analysis techniques along with continuous improvement to provide comprehensive protection against attacks. The framework was evaluated through experiments on various web applications, and the results showed that it was effective in detecting and preventing vulnerabilities such as SQL injection attacks and authentication leaks.

The methodology proposed in this paper is unique in that it not only analyzes the source code of web applications but also modifies it if necessary to eliminate vulnerabilities. Additionally, the continuous improvement aspect ensures that the framework is up-to-date with new types of attacks and vulnerabilities.

Overall, the proposed framework provides an effective solution to the increasing problem of web application security. By incorporating the methodology into the development process, web application developers can ensure that their applications are protected against the latest threats and vulnerabilities.

Future work includes further testing and refinement of the framework, as well as the development of new techniques for identifying and preventing attacks. With the ever-increasing importance of web applications in today's society, it is crucial to continue to improve their security to ensure the safety and privacy of users.

8. REFERENCES

[1]. Shetty, S., & D'Souza, S. (2020). Web Application Security: A Systematic Literature Review. International Journal of Scientific & Engineering Research, 11(6), 1-7.

[2]. Bala, S. (2020). A Review of Web Application Security Challenges and Solutions. International Journal of Computer Science and Mobile Computing, 9(3), 129-137.

[3]. Akhter, F., & Khan, M. U. (2019). A Comprehensive Review of Web Application Security Techniques. Journal of Information Security, 10(1), 1-20.

[4]. Zhou, M., & Luo, X. (2019). A Review of Web Application Security Research. Journal of Cyber Security Technology, 3(1), 1-8.

[5]. Patel, N., Patel, D., & Patel, H. (2020). A Study on Web Application Security Threats, Attacks and Prevention Techniques. Journal of Computer Science, 16(1), 62-76.

[6]. Sethi, P., & Sharma, S. K. (2019). A Survey on Web Application Security: Vulnerabilities, Attacks, and Prevention Techniques. International Journal of Computer Science and Information Technology Research, 7(2), 120-128.

[7]. Liu, Y., Han, G., & Ma, J. (2020). Web Application Security Assessment: A Survey. Journal of Information Security and Applications, 51, 1-17.

[8]. Kalia, N., & Yadav, P. (2019). A Survey of Web Application Security Attacks and Countermeasures. International Journal of Advanced Research in Computer Science, 10(4), 131-136.

[9]. Sharif, M., & Sultana, S. (2020). A Review on Web Application Security and Vulnerabilities. International Journal of Computer Sciences and Engineering, 8(7), 74-81.

[10]. Chaudhary, R., Sharma, N., & Kumar, N. (2020). A Comprehensive Review of Web Application Security. International Journal of Advanced Computer Science and Applications, 11(7), 252-257.