Global Scientific JOURNALS

GSJ: Volume 9, Issue 10, October 2021, Online: ISSN 2320-9186
www.globalscientificjournal.com

JSON INJECTION VULNERABILITIES

Mohammad

Department of Computer Science & Information Technology, Univerity of Engineering & Techology Peshawar, Pakistan
Email: mohammadnh93@gmail.com, sadeeqjan@yahoo.com

ABSTRACT

Modern business organization uses JSON data exchange format for communication purposes between backend systems (such as RESTful web services or RESTful APIs) and frontend systems (such as web applications). Client uses a frontend system to access different organization services in a sophisticated manner without knowing about the internal mechanism. If these frontend systems are vulnerable to JSON injection vulnerability, then it compromises the organizational security from sensitive data retrieval to system damage. JSON injection can be exploited by entering malicious input strings from the frontend system (HTML login form) into the backend system without proper validation and sanitization of incoming JSON data. Therefore, frontend security testing is important to protect the backend systems from destruction. In this research, we have done an analysis of JSON-based injection vulnerabilities in web applications and services and recommended to design an effective approach to detect this type of vulnerability by using security testing techniques.

Keywords: Security Testing, Injection Vulnerabilities, JSON Injection, White Box Testing, Black Box Tesing, RESTful APIs, Web Security

INTRODUCTION

In the fast-growing era of cyberspace, every business is encountering a persistent cyber threat. They are being victimized by a variety of sources. No business can guarantee 100 percent protection from cyber-attacks. In short, cyberspace is threatened far more than one could possibly imagine. The claim was report back in 2017 by Kaspersky Labs that their lab detection technology receives 360,000 new malicious files per day. This statistic shows a rise of 11.5% from last years and malware is one of the most threats facing by businesses [1]. According to 2019's acunetix report, 46% of websites contain high risk vulnerabilities while 87% of websites containing medium risk vulnerabilities [2]. The detection and fixing of a vulnerability are the best way to improve the protection of all businesses in order to avoid further exploitation.

Injection vulnerabilities are the most active attack vectors that permit an attacker to inject malicious code into a system using a web application (front-end systems). In simple terms, when malicious inputs are accepted by the front-end systems and permit these malicious inputs to enter the back-end systems such as web services, databases, etc. by making the web applications vulnerable to an injection flaw. Injection flaws in a system can result data loss, denial of service (DoS), loss of data integrity, and complete system compromise. Injection vulnerability is a major concern in web security therefore, it is listed as the number-one web application vulnerability in the Open Web Application Security Project (OWASP) Top 10, 2017 [3].
Predominantly, when computer geeks hear about Injection vulnerability, they might think about SQL injection, which is a common

injection form, but it is far away from the whole picture. However currently, the most influential type of injection is known as JSON injection. JSON injections are not very common as many other vulnerabilities, but they can lead to other dangerous injection attacks such as SQL Injection, NoSQL injection, API injection, XML injection, and Cross-Site Scripting (XSS) or other vulnerabilities like HTTP request smuggling [4].

BACKGROUND

What is JSON?

JavaScript Object Notation (JSON) is lightweight data exchange format which is used by web services to communicate with other systems. Before JSON, XML was used for data communication, but XML is more verbose and complex than JSON.

JSON Injection

JSON injection is an injection attack that can be found in a system using JSON data exchange for data communication. These attacks can be exploited when someone changes or inject a malicious string in the JSON data parameter and then send or received without validation.

Types of JSON injection attacks:

There are two types of attacks

i. Client-Side attack

JSON injection attacks on the client-side take place when JSON data are parsed with the eval () function.

ii. Server-Side attack

JSON injection attack take place on the server-side when JSON data are not sanitized by the server in a proper way.

JSON Injection a carrier

JSON injection can exploit other injection vulnerabilities. Some are listed below:

1. SQL Injection (SQLi)
2. XML Injection (XMLi)
3. Cross Site Scripting (XSS)
4. JSON Hijacking
5. Cross Site Request Forgery (CSRF)
6. XML External Entity Injection (XXE)

Security Testing for Front-End systems

There are two most common security testing methods to test frontend systems.

i. Black Box

In this method, frontend systems are tested without knowing their internal structure (code). The system is tested blindly via multiple inputs to check their effect on systems.

ii. White Box

In this method, frontend systems are tested with knowing their internal structure (code). Each path and coverage of the internal system are tested carefully to find any faults.

LITERATURE REVIEW

The purpose of security testing is to discover vulnerabilities in the software systems and ensure that it is secure from any vulnerabilities, so it does not exploit or stop functioning. Vulnerabilities not detected in the various testing stages could have significant implications vary from client frustration to harm of physical property or even to the risk of human lives [5]. It is therefore important to test advanced software systems thoroughly. Security testing uses the black box and white box testing techniques for the evaluation of vulnerabilities in software systems [6].

CONCLUSION

JSON injection vulnerability is an injection attack type that can be exploited by injection malicious string from input field of front-end system to backend systems. This vulnerability can become a carrier for other injection vulnerabilities such as SQL injection, XML injection, XSS, etc. JSON injection vulnerability not detected on time can cause some serious damage to the organizational infrastructure. An effective approach is required to detect JSON injection vulnerability in web applications and services.

REFERENCES

[1] Kaspersky, “Kaspersky Lab.” https://www.kaspersky.com/about/press-releases/2017_kaspersky-lab-detects-360000-new-malicious-files-daily (accessed Feb. 02, 2021).

[2] Acunetix, “Acunetix Web Application Vulnerability Report 2019.” https://www.acunetix.com/blog/articles/acunetix-web-application-vulnerability-report-2019/ (accessed Feb. 02, 2021).

[3] OWASP, “OWASP Top Ten.” https://owasp.org/www-project-top-ten/ (accessed Feb. 02, 2021).

[4] J. Clarke-Salt, SQL injection attacks and defense. Elsevier, 2009

[5] J. Wegener, “Evolutionary Testing Techniques BT - Stochastic Algorithms: Foundations and Applications,” 2005, pp. 82–94.

[6] S. Jan, C. D. Nguyen, A. Arcuri, and L. Briand, “A Search-Based Testing Approach for XML Injection Vulnerabilities in Web Applications,” in Proceedings - 10th IEEE International Conference on Software Testing, Verification and Validation, ICST 2017, May 2017, pp. 356–366, doi: 10.1109/ICST.2017.39.