International Journal of Digital Crime and Forensics
Volume 14 • Issue 2

Web Vulnerability Detection Analyzer Based on Python

Dawei Xu, Changchun University, China*
https://orcid.org/0000-0003-4422-0606

Tianxin Chen, Changchun University, China

Zhonghua Tan, Hainan Normal University, China

Fudong Wu, Changchun University, China

Jiaqi Gao, Changchun University, China

Yunfan Yang, Changchun University, China

ABSTRACT

In the information age, hackers will use web vulnerabilities to infiltrate websites, resulting in many security incidents. To solve this problem, security-conscious enterprises or individuals will conduct penetration tests on websites to test and analyze the security of websites, but penetration tests often take a lot of time. Therefore, based on the traditional web vulnerability scanner, the web vulnerability detection analyzer designed in this article uses vulnerability detection technologies such as sub-domain scanning, application fingerprint recognition, and web crawling to penetrate the website. The vulnerability scanning process of the website using log records and HTML output helps users discover the vulnerability information of the website in a short time and patch the website in time. It can reduce the security risks caused by website vulnerabilities.

KEYWORDS

Fingerprint Recognition, Network Security, Penetration Test, Subdomain Scanning, Vulnerability Scanning, Web Crawler, Web Security, Web Vulnerability

1. INTRODUCTION

With the continuous emergence of advanced Web application technologies in the Internet era, related Web vulnerabilities are also emerging. Web vulnerabilities may be due to lack of consideration of web security by website developers when developing websites, resulting in related security vulnerabilities in applications. Common web security vulnerabilities include SQL injection vulnerabilities, cross-site scripting vulnerabilities, and cross-site request forgery vulnerabilities. etc. (Yang Guofeng. 2019). Hackers can conduct penetration tests on target websites and use Web vulnerabilities to escalate privileges on website servers to achieve the purpose of invading websites. Based on these security threats, there is some value in using vulnerability scanners to detect vulnerabilities on websites.

DOI: 10.4018/IJDCF.302875

*Corresponding Author

The scanning process of traditional scanners is generally to obtain the URL of the website through a crawler, send a request with attack parameters to the website to obtain the payload, and output the corresponding vulnerability report if the payload is successfully verified. If the verification fails, continue to send the next request. Due to the high concurrency between modules, the next task can only be started after the completion of the previous task. The Web vulnerability detection analyzer designed in this paper can collect website information in batches to achieve high concurrency between modules, and tasks can be processed between crawlers and plug-ins at the same time, improving the efficiency of scanning websites, and the vulnerability script of the system has Scalability is conducive to the improvement and upgrade of the system. The vulnerability detection analyzer adopts a callable plug-in framework, which can automate the scanning process, actively send a request with parameters to the target website, and detect website vulnerabilities according to the obtained response.

Contributions made in this paper include:

1. According to the process analysis of website vulnerability scanning, the overall architecture of the web vulnerability detection analyzer and the functional requirements of the four modules are designed.
2. According to the cross-platform operation requirements of vulnerability scanning, the system is written in Python language.
3. According to the requirements of vulnerability verification, this paper uses a custom PoC plug-in to verify website vulnerabilities, uses multi-process concurrent engine operation mode, uses logs to record the response information returned by website requests, and provides targets for vulnerability verification.
4. For the completed system, the vulnerability scanning test of the website is carried out to test whether the function of the system is complete and the efficiency of scanning website vulnerabilities. This paper conducts vulnerability scanning tests on hundreds of websites, and divides these websites into three different scales. Test the total scan time of the website and the accuracy of website vulnerability results.

2. BACKGROUND

The essence of Web application security problems stems from the quality of software. Compared with traditional software, web applications are usually considered to be enterprise-specific applications, and functions in them need to be changed frequently to maintain normal business, which leads to a longer development cycle for web applications; due to the communication between the client and the server. The process is cumbersome, and it is not easy for many development technicians to sort out the communication logic, which leads to problems in the security of Web applications.

For web application developers, common vulnerabilities in web security arise when developing web applications. Most programmers who develop websites have weak security awareness. They trust any data entered by users when writing application code. The input parameters are not tested or strictly tested, and the relevant technical personnel do not consider the filtering of special characters. This means that many applications have web vulnerabilities (Teng Fei. 2020). In the process of writing code, the programmer may use the interface incorrectly or the interface is not perfect, and there may be defects in the code function or logic loophole in the application program. Many web vulnerabilities can be avoided if the web application code takes into account the security of the website. Therefore, web application developers should strictly identify and filter the special characters of website HTTP headers, keyword queries and POST data entered by users in the process of code development. For web site administrators, failure to configure the type of software correctly, to patch vulnerable sites in a timely manner, and to properly handle abnormal problems in web applications are the main reasons for web security problems. Web site administrators can scan for vulnerabilities to identify the configuration parameters of various sites, install the latest security patches on the sites in a timely