Analyses of Security Scanning and Security Threat in Web Application Network

김정태
목원대학교

웹에서의 보안 위협과 시큐리티 스캐닝에 대한 분석

Jung-Tae Kim
Mokwon University
E-mail : jtkim5068@gmail.com

요 약

In this paper, we analyses a testing methodology that allows for harmless auditing, define three testing modes heavy, relaxed, and safe modes, and report our results from two experiments. In the first, we compared the coverage and side effects of the three scanning modes using Web applications chosen from the different vulnerable in a previous static verification effort.

Ⅰ. Introduction

There are a number of studies performing large-scale search query analysis. However, the number of studies analyzing the Web searching within a time-based frame is limited. The current comparative study provides a time-based analysis including data from the Excite and Fast Web search engines. Software and hardware limitations gain importance in handling databases of millions of queries. Moreover, many studies on Web user query sessions require context-wise interpretation of data that require manual analysis. Researchers have proposed a broad range of defense strategies against XSS attacks. Park and Sandhu's cookie-securing mechanism can be adopted to eliminate XSS, but it requires explicit modifications to existing Web applications. Scott and Sharp have proposed using gateways for filtering malicious input at the application level. In addition to preventing XSS, the gateway also prevents SQL injection another widespread Web application vulnerability. [1]

II. Definition of Web Application Scanner

A web application scanner is an automated program that examines web applications for security vulnerabilities. In addition to searching for web application specific vulnerabilities, the tools also look for software coding errors, such as illegal input strings and buffer overflows. Web application scanner explores an application by crawling through its web pages and performs penetration testing, an active analysis of a web application by simulating attacks on it. This involves generation of malicious inputs and subsequent evaluation of application's response. Web application scanner performs different types of attack.

A generally useful attack, called fuzzing, is submitting random inputs of various sizes to the application. Penetration testing is a black-box testing approach. The limitation of this approach is its inability to examine source code, thus it is unlikely to detect such vulnerabilities as back doors. However, it is well suited for detecting input validation problems. Additionally, client-side code is available to the penetration tester and can provide important information about the inner workings of a Web application.[2]

III. Web Application Vulnerability Process

The primary objectives of information security systems are to protect confidentiality, integrity, and availability. From our examples, it is obvious that for Web applications, compromises in integrity are the main causes of compromises in confidentiality and availability. The relationship is illustrated in Figure 1. When untrusted data is used to construct trusted output without sanitization, violations in data integrity occur, leading to escalations in access rights that result in availability and confidentiality compromises.[1] We made the following assumptions when designing our approach to detecting XSS and SQL injection vulnerabilities:

Assumption 1: All data sent by Web clients in the form of HTTP requests should be considered untrustworthy.

Assumption 2: All data local to a Web application are secure.

Assumption 3: Tainted data can be made secure(against known attacks) with appropriate processing.

In addition, the following policies were defined:

Policy 1: Tainted data must not be used in HTTP response construction.

Policy 2: Tainted data must not be written into local Web application storage.

Policy 3: Tainted data must not be used in system command construction.

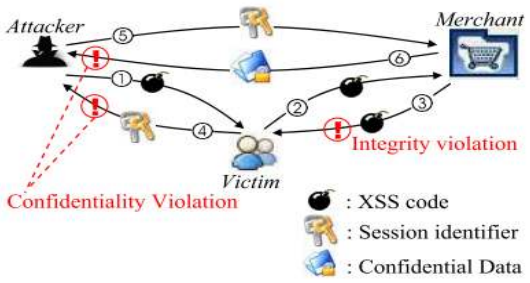

Fig.1 Configuration of Web Application Vulnerability

IV. Conclusion

In researchers efforts to assess web application security, researchers from both academic and private sector are devoting a considerable amount of resources to developing web application security scanners. They are achieving some success, but little is known about potential side effects. In this paper, we analyses web application vulnerability process and threats.

References

[1] Yao-Wen Huang, etcs, "Non-Detrimental Web Application Security Scanning" Proceedings of the 15th International Symposium on Software Reliability Engineering(ISSRE'04)

[2] Huang, Y. W., Huang, S. K., Lin, T. P., Tsai, C. H. "Web Application Security Assessment by Fault Injection and Behavior Monitoring." In Proc. 12th Int'l World Wide Web Conference, p.148-159, Budapest, Hungary, 2003.

[감사의 글]

본 논문은 2010년도 한국과학기술단체총연합회의 지원을 받아 "이공계전문가기술지원서포터즈 기업 및 맞춤형멘토지원사업"에서 수행된 기초연구사업입니다.