

Chapter 9

Enhancing the Browser–Side Context–Aware Sanitization of Suspicious HTML5 Code for Halting the DOM–Based XSS Vulnerabilities in Cloud

B. B. Gupta

National Institute of Technology Kurukshetra, India

Shashank Gupta

National Institute of Technology Kurukshetra, India

Pooja Chaudhary

National Institute of Technology Kurukshetra, India

ABSTRACT

This article presents a cloud-based framework that thwarts the DOM-based XSS vulnerabilities caused due to the injection of advanced HTML5 attack vectors in the HTML5 web applications. Initially, the framework collects the key modules of web application, extracts the suspicious HTML5 strings from the latent injection points and performs the clustering on such strings based on their level of similarity. Further, it detects the injection of malicious HTML5 code in the script nodes of DOM tree by detecting the variation in the HTML5 code embedded in the HTTP response generated. Any variation observed will simply indicate the injection of suspicious script code. The prototype of our framework was developed in Java and installed in the virtual machines of cloud environment on the Google Chrome extension. The experimental evaluation of our framework was performed on the platform of real world HTML5 web applications deployed in the cloud platform.

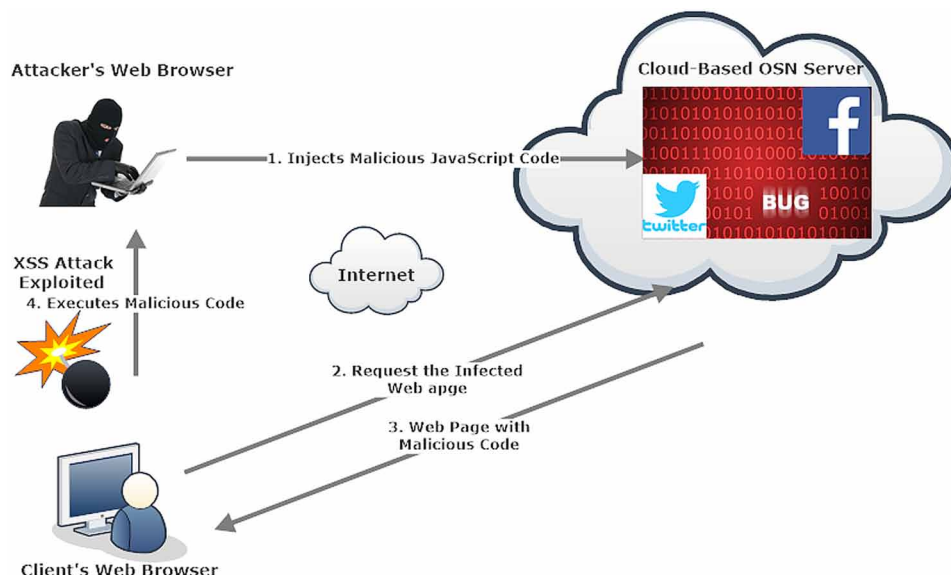
DOI: 10.4018/978-1-5225-3422-8.ch009

INTRODUCTION

The tremendous explosion in cloud computing produced numerous security issues related to data security of cloud users (Dinh et al., 2013; Gupta et al., 2016c). The propagation of XSS worms are considered to be topmost threat originated in HTML5 web applications deployed in the framework of cloud infrastructure. In the contemporary era of cloud computing, cloud security has turned out to be a serious issue, as numerous on-demand resources are being offered by utilizing the virtualization technologies of cloud services (Modi et al., 2013). Instead of referring the outdated Internet settings for constructing an expensive setup, numerous commercial IT organizations are accessing the services of Online Social Networking (OSN) sites (such as Twitter, Facebook, LinkedIn, etc.) on the cloud platforms. In the modern era of Web 2.0 technologies and HTML5-based web applications, OSN is considered to be the most popular method for information sharing has drawn most of public attention. However, it is clearly known that the cloud settings are installed on the backbone of Internet. Therefore, numerous HTML5 web application vulnerabilities in the conventional Internet infrastructures also exist in the backgrounds of cloud-based environments.

The most prominent attack found on HTML5 web applications is the Cross Site Scripting (XSS) attack [Gupta et al. (2016a), Gupta et al. (2016b), Gupta et al. (2015a), Gupta et al. (2015b), Gupta et al. (2014)]. Figure 1 highlights the injection of XSS worm on the OSN web server deployed in the virtual machines of cloud platforms. XSS worms have turned out to be a plague for the cloud-based HTML5 web applications. Such worms steal the sensitive credentials of the active users by injecting the malicious HTML5 script code in the form of some posts on such web applications [Gupta et al. (2015c), Duchene et al. (2014), Shahriar et al. (2011), Doupe et al. (2013), Chandra et al. (2011), Xiao et al. (2014)]. Input sanitization is considered to be the most effective mechanism for alleviating and mitigating the effect of XSS worms from the cloud-based HTML5 web applications on the virtual machines of cloud platforms.

Figure 1. Exploitation of XSS attack on cloud platform



30 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/enhancing-the-browser-side-context-aware-sanitization-of-suspicious-html5-code-for-halting-the-dom-based-xss-vulnerabilities-in-cloud/188209

Related Content

Model Driven Approach to Secure Optimized Test Paths for Smart Samsung Pay using Hybrid Genetic Tabu Search Algorithm

Nisha Rathee and Rajender Singh Chhillar (2018). *International Journal of Information System Modeling and Design* (pp. 77-91).

www.irma-international.org/article/model-driven-approach-to-secure-optimized-test-paths-for-smart-samsung-pay-using-hybrid-genetic-tabu-search-algorithm/208640

The Recovery Language Approach

Vincenzo De Florio (2009). *Application-Layer Fault-Tolerance Protocols* (pp. 175-241).

www.irma-international.org/chapter/recovery-language-approach/5126

A Survey of Software Architecture Approaches

Kendra M.L. Cooper, Lirong Dai, Renee Steiner and Rym Zalila Mili (2009). *Designing Software-Intensive Systems: Methods and Principles* (pp. 256-288).

www.irma-international.org/chapter/survey-software-architecture-approaches/8239

Enterprise Management Data Acquisition System Based on WoT

Kun Yan and Dongyan Li (2022). *International Journal of Information System Modeling and Design* (pp. 1-11).

www.irma-international.org/article/enterprise-management-data-acquisition-system-based-on-wot/313577

Examining Open Source Software Licenses through the Creative Commons Licensing Model

Kwei-Jay Lin, Yi-Hsuan Lin and Tung-Mei Ko (2009). *Software Applications: Concepts, Methodologies, Tools, and Applications* (pp. 2978-2990).

www.irma-international.org/chapter/examining-open-source-software-licenses/29546