Chapter 42
Information Theoretic XSS Attack Detection in Web Applications

Hossain Shahriar
Kennesaw State University, USA

Wei-Chuen Chen
Kennesaw State University, USA

Sarah North
Kennesaw State University, USA

Edward Mawangi
Kennesaw State University, USA

ABSTRACT

Cross-Site Scripting (XSS) has been ranked among the top three vulnerabilities over the last few years. XSS vulnerability allows an attacker to inject arbitrary JavaScript code that can be executed in the victim's browser to cause unwanted behaviors and security breaches. Despite the presence of many mitigation approaches, the discovery of XSS is still widespread among today's web applications. As a result, there is a need to improve existing solutions and to develop novel attack detection techniques. This paper proposes a proxy-level XSS attack detection approach based on a popular information-theoretic measure known as Kullback-Leibler Divergence (KLD). Legitimate JavaScript code present in an application should remain similar or very close to the JavaScript code present in a rendered web page. A deviation between the two can be an indication of an XSS attack. This paper applies a back-off smoothing technique to effectively detect the presence of malicious JavaScript code in response pages. The proposed approach has been applied for a number of open-source PHP web applications containing XSS vulnerabilities. The initial results show that the approach can effectively detect XSS attacks and suffer from low false positive rate through proper choice of threshold values of KLD. Further, the performance overhead has been found to be negligible.

1. INTRODUCTION

Vulnerabilities are frequently discovered in web applications. Among all known vulnerabilities, Cross-Site Scripting (XSS) has been ranked among the top three vulnerabilities over the last few years (OWASP 2013). XSS vulnerability opens up the possibility for an attacker to inject arbitrary JavaScript code (OWASP-XSS 2013) that can execute in the context of a victim's browser. The injected script code causes

DOI: 10.4018/978-1-5225-3422-8.ch042

unwanted behaviors (*e.g.*, generating pop up windows) and security breaches (*e.g.*, session hijacking (Msujaws 2011)). A recent survey also shows that on average 60% or more web applications are currently suffering from XSS vulnerabilities (Tudor 2013). Given that statistic, addressing the mitigation of XSS vulnerabilities is important.

Despite the presence of many mitigation approaches for XSS attacks at both client and server sides (Shar *et al.* 2012; Frenz *et al.* 2012; Jim *et al.* 2007; Kirda *et al.* 2006; Iha *et al.* 2009; Gundy *et al.* 2009; Wurzinger *et al.* 2009), the discovery of XSS vulnerability is still widespread among today's web applications. Most of these approaches rely on signature-based attack detection that are effective in detecting known attack symptoms. Thus, there is a need to develop anomaly-based attack detection techniques that may detect unknown and new attack signatures. This paper applies an information theoretic concept to detect XSS attacks. Further, very few works have explored detecting XSS attacks at the proxy level.

In this article, we propose a proxy-level XSS attack detection technique based on a popular information theoretic measure known as Kullback-Leibler Divergence (KLD)[1]. Our intuition is that legitimate JavaScript code present in web applications should remain similar or very close to the JavaScript code of a rendered web page. A high deviation between the two set of JavaScript code may indicate XSS attacks. Our contribution remains in addressing the missing elements when computing KLD between the set of expected and actual JavaScript code. In particular, we apply the constant back-off smoothing technique that we brought from information retrieval literature.

We apply the proposed XSS attack detection approach for web applications implemented in PHP language and containing known XSS vulnerabilities. The initial results show that the approach can detect most of the known XSS attack signatures and show negligible false positive warning. Further, it imposes negligible runtime overhead. The proposed approach can handle diverse types of JavaScript code commonly found in web applications such as inline, URL attribute, and Cascading Style Sheet (CSS). Moreover, it can be applied as a complementary defense technique for applications that may lack an adequate XSS input filtering mechanism.

This article is organized as follows: First, we show an example of XSS attack followed by a brief introduction of related work. Then the proposed KLD-based XSS attack detection framework is discussed along with a working example. We then discuss the experimental results. Finally, we conclude the paper and discuss future work.

1.1. An Example of XSS Attack

Figure 1(a) shows the HTML code of a web page that accepts the user name in an HTML form. Here, a user can supply his/her name in a text box for displaying the name in PHP script. The supplied input is accessed in *show.php* shown in Figure 1(b). Note that *$_POST["user"]* variable retrieves the supplied input and then displays it in the response page (*echo* statement). The displayed content is not being filtered for possible JavaScript code. As a result, if a malicious user supplies arbitrary JavaScript code, it would execute when the response page is displayed in the victim's browser.

For example, if the user name is supplied as <*script*>*alert (document.cookie)*</*script*>, then browser will display the cookie information. This is an example of reflected XSS attack. The other common variant is known as stored XSS attack where injected payloads are stored at the server side storage system. The attack payloads are retrieved at a later time when a victim sends a page request at the client side.

Note that XSS attacks bypass the default Same Original Policy (SOP) in browsers which is intended to prevent accessing one webpage downloaded from a domain to a page from another domain. For ex-