Hindawi
Wireless Communications and Mobile Computing
Volume 2017, Article ID 5360472, 9 pages
https://doi.org/10.1155/2017/5360472

Research Article

Defending Malicious Script Attacks Using Machine Learning Classifiers

Nayeem Khan, Johari Abdullah, and Adnan Shahid Khan

Department of Computer Systems & Communication Technologies, Faculty of Computer Science & Information Technology, Universiti Malaysia Sarawak, 94300 Kota Samarahan, Sarawak, Malaysia

Correspondence should be addressed to Nayeem Khan; 15010049@siswa.unimas.my

Received 27 October 2016; Accepted 29 December 2016; Published 7 February 2017

Academic Editor: Paul Honeine

The web application has become a primary target for cyber criminals by injecting malware especially JavaScript to perform malicious activities for impersonation. Thus, it becomes an imperative to detect such malicious code in real time before any malicious activity is performed. This study proposes an efficient method of detecting previously unknown malicious java scripts using an interceptor at the client side by classifying the key features of the malicious code. Feature subset was obtained by using wrapper method for dimensionality reduction. Supervised machine learning classifiers were used on the dataset for achieving high accuracy. Experimental results show that our method can efficiently classify malicious code from benign code with promising results.

1. Introduction

Due to the rapid advancement of modern computers and network infrastructure and our dependence on the Internet and its services which are increasing, web applications which are accessed through web browsers have become a primary target for cyber criminals. As per a report released by Symantec in 2016 [1], about 430 million unique pieces of malware were discovered showing a growth of 36% against the year 2014. Cyber criminals use the malicious code and malicious URLs to attack individuals and organizations. The sole purpose of such attacks is for personal, financial, and political gains through damaging or disrupting normal operation of computer and systems, performing phishing attacks, displaying unwanted advertisements, and extorting money. Thus detection of such malicious code attacks becomes a topmost security challenge.

Open Web Application Security Project (OWASP) has ranked cross-site scripting (XSS) as the 2nd most dangerous vulnerability among top ten vulnerabilities. Currently, XSS holds a share of 43% among all the reported vulnerabilities. XSS is a type of injection attack in which malicious scripts are injected around benign code in a legitimate webpage in order to access cookie, session, and other secret information. Web applications are used to transport malicious scripts to perform the attack. The target of XSS attack is a client side whereas SQL injections target server-side [2]. XSS attack is a vulnerability at the application layer of network hierarchy, which occurs by injecting malicious scripts to break security mechanism. About 70% attacks are reported to occur at application layer. Web browsers are the most susceptible application layer software for attacks. The purpose of the web browser is to get the requested web resource from the server and displayed in browser's windows. The format of the supplied resources is not restricted to HyperText Markup Language (HTML) but can also be portable document format (PDF), image, and so on. Attackers run malicious JavaScript in a web browser to target users. Malicious and obfuscated URLs also serve as a carrier for XSS attacks [3].

JavaScript is a programming/scripting language, used in web programming for making web pages more interactive, adds more features, and improves the end user experience. JavaScript also helps in reducing the server-side load and helps in shifting some computation to end user side. The JavaScript is commonly used for client-side scripting to be used in web browsers. When a user requests for a certain web page through a web browser, the server responds to the request and sends back web page which might have