PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

UNIVERSITY OF M'SILA
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE

COMPUTER SCIENCE DEPARTMENT

Dissertation Submitted in partial fulfillment of the requirements for the Degree of MASTER

Domain: Mathematics and Computer Science

Branch: Computer Science

Specialty: Networks

By: GHERBI ADEL AMINE

TOPIC

Design and Development of Anti-XSS Proxy

Publicly defended: / /2016 before a Jury composed of :

Ms. SAOUDI LALIA	**University of M'sila**	**Supervisor**
.....	**University of M'sila**	**Chair**
.....	**University of M'sila**	**Examiner**
.....	**University of M'sila**	**Examiner**

Academic Year: 2015 /2016

Table of content

General introduction

Context

Nowadays, with the network expanding quickly, internet is not anonymous to us anymore. It is a good platform for users to communicate, chat, do business or play game together. For instance, we usually go to Google to search information, Amazon or E-Bay to buy books and many other goods and we also go to My-Space to communicate with friends. Therefore, there is no doubt that Internet is gradually becoming an integral part our daily life.

So providing a beneficial and safe networking environment is significantly necessary. If there is vulnerability in a famous website, many visitors will be attacked customers and the result cannot be imagined. Ten years ago, most of the websites were static websites which did not have too much vulnerability and were not interactive with visitors so that they could not be spitefully used by hackers and we ignored the WEB-based security. However, today there are millions and millions of dynamic websites with a lot of new technology being carried out and used into web browser. There are many plug-in applications which increase the interaction between visitors, for example, e-mail forms to provide the user interaction with the web server.

Statement of the Problem

However, everything has two sides. On the opposite side, these dynamic websites also provide a good platform for hackers to inject malicious code, as well. If the code is executed behind the web browser, it changes the web page according to the code automatically. Therefore, we find that a lot of famous websites were injected with malicious code by hackers and a lot of visitors were attacked. Moreover, owing to the extensive spread of Web 2.0 and each user's blog can be shared with his/her friends as well.

So, if one blog has been injected with malicious code, all the visitors of the blogger's friends will be infected and constantly infect their friends. Therefore, the speed of spreading is even quicker than previously. Eventually, the website provider will lose a lot of money and its reputation will be damaged, as well. Cross-Site-Script (XSS) vulnerability is one of those vulnerabilities. "XSS carried out on websites was roughly 80% of all documented security vulnerabilities as of 2007 [5] and it can let hackers insert the malicious JavaScript into a website. So the visitor of the website will be attacked and execute the malicious code automatically.

In addition, it can steal visitors' cookies and acquire the visitor's right as well. Therefore, it threatens the web security in the client part directly and steals the visitors' information catlike or redirects the visitors to visit another website which the hacker has established with malicious code already. It even can control the visitor's computer.

Although there are some methods which can detect XSS attacks or threats, XSS still cannot be completely detected. AS XSS code can be flexibly constructed, it is a significant problem and is also used to attack a lot of famous website, like: Yahoo, Myspace, Joomla-based websites and so on. Therefore, we have to pay more attention to this vulnerability and find out more methods to prevent these attacks and this research paper will explain the details of this vulnerability in order to make more people be aware of it.

General Objectives

The subject of this work is to develop a prototype tool that detects XSS attack using a dynamic way. The main goal is to improve the detection of the XSS attack and protect the server side and the client side.

Methodology

Our approach is divided in two phases: training phase and detection phase

The Training phase represent the understanding and analysis step in the reverse proxy , it has five modules: crawler, database, script extractor, XSS grammar, and script hasher.

The detection phase receives client requests and the response page and analyses them. This phase has six modules: Request parameters extractor, Sanitization, Response page analyzer, Script hasher, Compare hash and XSS type detection.

Report Outline

Chapter one deals mainly with the web applications and the HTTPs and gives an idea about the web securing and talks about XSS and its classifications, some attack techniques, and its impact on the web pages

Chapter two offers a brief History of Intrusion Detection Systems, and gives a main ideas about IDS and its types and architecture. In addition Approaches to Intrusion Detection. Alongside with XSS attack detection techniques.

Chapter three define our approach of XSS attack detection, which uses a reverse proxy at the server side to intercepts any request from clients, and analyzes any response from the server before send it to the client.

Chapter four represents a full experiment of our anti-XSS solution and put a test to our system and finally the results obtained.

CONCLUSION AND PERSPECTIVES

With the XSS vulnerability gradually evolving, a lot of new bypassing filters expressions appear so that XSS vulnerability will become more and more popular and XSS attacks will increase, as well.

In this project we proposed a anti-XSS solution which works on web server reverse proxy,

Our approach is divided in two phases: training phase and detection phase:

The Training phase represent the understanding and analysis step in the reverse proxy , it has five modules: crawler, database, script extractor, XSS grammar, and script hasher.

The detection phase receives client requests and the response page and analyses them. This phase has six modules: Request parameters extractor, Sanitization, Response page analyzer, Script hasher, Compare hash and XSS type detection.

Our solution can help to detect the stored XSS and mitigate the potential damage that could be unleashed by a bit of malicious XSS code slipping the a Web application's input validation and escaping defenses by providing an early warning, but is still at a stage where much more rigorous testing needs to be applied to it to see how well it detects XSS attacks against the breadth of all possible XSS attacks on a diversity of different Web pages,

As perspective we hope to finish what we started by integrating our work with a reverse proxy. In order to get better results and realized in daily lives to increase security level.

Bibliography

[1] Jia, X. "Design, Implementation and Evaluation of an Automated Testing Tool for Cross-Site Scripting Vulnerabilities." Yüksek Lisans Tezi, Darmstadt University of Technology (TUD)-Computer Science Department 2.6 (2006).

[2] Justin, Clarke. "SQL Injection Attacks and Defense Second Edition". USA : Elsevier, 2012. 978-1-59749-963-7

[3] Marcel Dekker. "Security of the Internet", Froehlich/Kent Encyclopedia of Telecommunications , New York, 1997.

[4] Chris Joscelyne. Information Management, AUSTRALIAN PROJECTS PTY LIMITED IT Security and Data Protection, 2005.

[5] The Open Web Application Security Project (OWASP), https://www.owasp.org, visited 14/02/2016

[6] (OWASP), https://www.owasp.org/index.php/Top_10_2013-Top_10, visited 14/02/2016

[7] Erwan Abgrall, "An Empirical Study of Browser's Evolution Impact on Security & Privacy". Télécom Bretagne; Université de Rennes 1, 2014.

[8] Grossman, J., Hansen, R., Petkov, P., Rager, A., & Fogie, S. (2007). "Cross site scripting attacks: XSS Exploits and defense". Syngress, Elsevier, Amsterdam

[9] Shende Dinesh Ankush, XSS Attack Prevention Using DOM based fitering API. National Institute of Technology Rourkela, Rourkela – 769 008, India 2014.

[10] Billy K Rios, Raghav Dube , kicking down the cross domain door. March 2007

[11] OWASP,Cross-site Scripting (XSS) https://www.owasp.org/index.php/Cross-site_Scripting_(XSS) , visited 14/02/2016

[12] OWASP XSS Filter Evasion Cheat Sheet

https://www.owasp.org/index.php/XSS_Filter_Evasion_Cheat_Sheet,

[13] R. Fielding, J. Gettys, , J. Mogul, , H. Frystyk, and T. Berners-Lee, "Hypertext Transfer Protocol --HTTP/1.1", RFC 2616, June 1999.

[14] "http tutorial",[Online]. Available: http://www.tutorialspoint.com/http/http_pdf_version.htm visited 26/03/2016.

[15] Programming notes , HTTP (Hyper Text Transfer Protocol), https://www.ntu.edu.sg/home/ehchua/programming/webprogramming/HTTP_Basics.html - visited 29/01/2016 .

[16] Cenzic vulnerability report 2013 http://info.cenzic.com/rs/cenzic/images/Cenzic-ApplicationVulnerability-Trends-Report-2013.pdf

[17] Mirante, Dennis, and Justin Cappos. "Understanding password database compromises." Dept. of Computer Science and Engineering Polytechnic Inst. of NYU, Tech. Rep. TR-CSE-2013-02 (2013).

[18] BACE, Rebecca et MELL, Peter. NIST special publication on intrusion detection systems. BOOZ-ALLEN AND HAMILTON INC MCLEAN VA, 2001 [19] Nicholas J.Puketza, Kui Zhang, Mandy Chung, Biswanath Mukherjee, Ronald A.Olsson . A Methodology for Testing Intrusion Detection Systems.

[19] Puketza, N. J., Zhang, K., Chung, M., Mukherjee, B., & Olsson, R. A. (1996). A methodology for testing intrusion detection systems. *Software Engineering, IEEE Transactions*.

[20] CHADOULI Youssouf, SAOUDI Lalia, "A New Feature Selection approach For Network Intrusion Detection Systems"

[21] Rafeeq Ur Rehman ,Intrusion Detection Systems with Snort: Advanced IDS Techniques Using Snort, Apache, MySQL, PHP, and ACID

[22] Burton, James D., C. Tate Baumrucker, and Ido Dubrawsky. Cisco security professional's guide to secure intrusion detection systems. Syngress Publishing, 2003.

[23] N. Ikemiya and N. Hanakawa, "A New Web Browser Including A Transferable Function to Ajax Codes", In Proceedings of 21st IEEE/ACM International Conference on Automated Software Engineering (ASE '06), Tokyo, Japan, (2006) September.

[24] D.E. Denning, "An Intrusion-Detection Model," IEEE Trans. Software Eng., vol. 13, no. 2, pp. 222-232, Feb. 1987.

[25] V. Malviya, S. Saurav, A.Gupta, On Security Issues in Web Applications through Cross Site Scripting (XSS), APSEC '13 Proceedings of the 2013 20th Asia-Pacific Software Engineering Conference (APSEC) - Volume 01, Pages 583-588

[26] Debar, Herve. "An introduction to intrusion-detection systems." Proceedings of Connect 2000.

[27] Jacob, B. (2011). Automatic XSS detection and Snort signatures/ACLs generation by the means of a cloud-based honeypot system (Doctoral dissertation, Edinburgh Napier University).

[28] :Engin Kirda, Christopher Kruegel, Giovanni Vigna, and Nenad Jovanovic. Noxes: A Client-Side Solution for Mitigating Cross Site Scripting Attacks. Security Track of the 21st ACM Symposium on Applied Computing (SAC 2006), Dijon, France, April, 2006

[29] : Omar Ismail, Masashi Etoh, Youki Kadobayashi, and Suguru Yamaguchi. A Proposal and Implementation of Automatic Detection/Collection System for Cross-Site Scripting

Vulnerability. Proceedings of the 18th International Conference on Advanced Information Networking and Application (AINA'04), IEEE, 2004

[30]: T.Jim , N.Swamy and M.Hicks, " Defending against Cross-Site Scripting Attacks with Browser-Enforced Embedded Policies,"Proc of the WWW,Banff,Alberta,May 2007,pp. 601-610.

[31]: H. Shahriar and M. Zulkernine, "S2XS2: A Server Side Approach to Automatically Detect XSS Attacks," Proc. of the 9th IEEE International Conference on Dependable, Autonomic and Secure Computing (DASC), Sydney, Australia, December 2011, pp. 7-14.

[32] M. Gundy and H. Chen, "Noncespaces: Using Randomization to Enforce Information Flow Tracking and Thwart Cross-site Scripting Attacks," Froc. of the 16'h Network & Distributed System Security Symposium, San Diego, February 2009.

[33]: P.wurzinger,C.Platzer,C.ludl,E.kirda and C.Kruegel, "SWAP:Mitigating XSS Attacks using Reverse Proxy, "Proc. Of the SESS,Vancouver,Msy 2009,pp. 33-39.

[34]: Hossain Shahriar, Sarah North1, Wei-Chuen Chen, and Edward Mawangi, Design and Development of Anti-XSS Proxy. ICITST- 2013.

[35] : C. Frenz, J. Yoon, "XSSmon: A Perl based IDS for the Detection of Potential XSS Attacks," Systems, Applications and Technology Conference (LISAT), Proc. of 2012 IEEE Long Island, May 2012, pp. 1 - 4.

[36]: G. Wassermann, and Z. Su, "Static detection of cross-site scripting vulnerabilities," Proceedings of the 30th international conference onSoftware engineering (ICSE '08), New York, USA, pp. 171-780, 2008.

[37]: N. Jovanovic, C. Kruegel, and E. Kirda, "Precise alias analysis for static detection of web application vulnerabilities," Proceedings of the 2006 workshop on Programming languages and analysis for security (PLAS '06), New York, USA, pp. 27-36, 2006.

[38]: Y. Wang, Z. Guo, "Program slicing stored XSS bugs in web application," Proceedings of the Fifth IEEE International Conference on Theoretical Aspects of Software Engineering, pp. 191-194, 2011.

[39]: N. Jovanovic, C. Kruegel and E. Kirda, "Pixy: A static analysis tool for detecting web application vulnerabilities (short paper)," In 2006 IEEE Symposium on Security and Privacy, Oakland, CA, (2006) May .

[40] Y. W Huang, F. Yu, C. Hang, C. H. Tsai, D. Lee and S. Y. Kuo, "Verifying Web Application using Bounded Model Checking," In Proceedings of the International Conference on Dependable Systems and Networks, (2004).

[41] Y.-W. Huang, S.-K. Huang, T.-P. Lin and C.-H. Tsai, "Web application security assessment by fault injection and Behavior Monitoring," In Proceeding of the 12th international conference on World Wide Web, ACM, New York, NY, USA, (2003).

[42] "Web Application Security Assessment," SPI Dynamics Whitepaper, SPI Dynamics, (2003).

[43] "Web Application Security Testing – AppScan 3.5", Sanctum Inc., http://www.sanctuminc.com.

[44] "InterDo Version 3.0", Kavado Whitepaper, Kavado Inc., (2003).

[45] Y.-W. Huang, F. Yu, C. Hang, C. H. Tsai, D. Lee and S. Y. Kuo, "Securing web application code by static analysis and runtime protection," In Proceedings of the 13th International World Wide Web Conference,(2004).

[46] D. Scott and R. Sharp, "Abstracting Application-Level Web Security," In Proceeding 11th international World Wide Web Conference, Honolulu, Hawaii, (2002).

[47] G. Wassermann and Z. Su, "Static detection of cross-site Scripting vulnerabilities," In Proceeding of the 30th International Conference on Software Engineering, (2008) May.

[48] A. S. Christensen, A. Møller and M. I. Schwartzbach, "Precise analysis of string expression", LNCS, Springer-Verlag, ISSN 0909-0878, February 2003.

[49] M. Mohri and M. Nederhof, "Regular approximation of context-free grammars through transformation", Robustness in Language and Speech Technology, (2001), pp. 153–163.

[50] Y. Minamide, "Static Approximation of Dynamically Generated Web Pages", In WWW'05: Proceedings of the 14th International Conference on the World Wide Web, (2005), pp. 432–441.

[51] T. Jim, N. Swamy and M. Hicks, "BEEP: Browser-Enforced Embedded Policies," In Proceedings of the 16th International World Wide Web Conference, ACM, (2007), pp. 601-610.

[52] P. Bisht and V. N. Venkatakrishnan, "XSS-GUARD: Precise dynamic prevention of Cross-Site Scripting Attacks," In Proceeding of 5th Conference on Detection of Intrusions and Malware & Vulnerability Assessment, LNCS, vol. 5137, (2008), pp. 23-43.

[53] Z. Su and G. Wassermann, "The essence of command Injection Attacks in Web Applications," In Proceeding of the 33rd Annual Symposium on Principles of Programming Languages, USA: ACM, (2006) January, pp. 372-382.

[54] D. Balzarotti, M. Cova, V. V. Felmetsger and G. Vigna, "Multi-Module Vulnerability Analysis of Webbased Applications," In proceeding of 14th ACM Conference on Computer and Communications Security,Alexandria, Virginia, USA, (2007) October.

[55] Anderson, J. P. (1980). Computer security threat monitoring and surveillance (Vol. 17). Technical report, James P. Anderson Company, Fort Washington, Pennsylvania.

[56]: N. Li, T. Xiea, M. Jin, and C. Liu, "Perturbation-based user-inputvalidation testing of web applications," The Journal of Systems and Software, vol. 83, pp. 2263-2274, 2010.

[57]: A. Nunan, E. Souto, E. M. dos Santos, and E. Feitosa, "Automatic classification of cross-site scripting in web pages using document based and URL based features," IEEE Symposium on Computers and Communications (ISCC), pp. 702-707, 2012.

[58] T. Pietraszek and C. V. Berghe, "Defending against Injection Attacks through Context-Sensitive String Evaluation", In Proceeding of the 8th International Symposium on Recent Advance in Intrusion Detection (RAID), (2005) September.

[59] Z. Su and G. Wassermann, "The essence of command Injection Attacks in Web Applications," In Proceeding of the 33rd Annual Symposium on Principles of Programming Languages, USA: ACM, (2006).

[60] D. Balzarotti, M. Cova, V. V. Felmetsger and G. Vigna, "Multi-Module Vulnerability Analysis of Webbased Applications," In proceeding of 14th ACM Conference on Computer and Communications Security, Alexandria, Virginia, USA, (2007) October.

[61] Y. Xie and A. Aiken, "Static detection of security vulnerabilities in scripting languages," Stanford University Stanford, CA 94305.

[62]"Forward Proxies and Reverse Proxies" , http://web.mit.edu/jhawk/mnt/spo/subversion/src/httpd-2.0/docs/manual/mod/mod_proxy.html

[63] Denning, Dorothy E., and Peter G. Neumann. "Requirements and model for IDES—a real-time intrusion detection expert system." Document A005, SRI International 333 (1985).

[64] Pant, Gautam, Padmini Srinivasan, and Filippo Menczer. "Crawling the web." *Web Dynamics*. Springer Berlin Heidelberg, 2004.

[65] www.levenshtein.net, visited 19-05-2016

[66] http://people.cs.pitt.edu/~kirk/cs1501/Pruhs/Fall2006/Assignments/editdistance/Levenshtein , visited 19-05-2016

[67] NetBeans, https://netbeans.org , visited 16-05-2016

[68] WampServer|sourceforge.net, https://sourceforge.net/projects/wampserver , visited 16-05-2016

[69] WampServer, http://www.wampserver.com , visited 16-05-2016

[70] MySQL, https://www.mysql.com , visited 16-05-2016

[71] Stubblebine, Tony. "Regular Expression Pocket Reference: Regular Expressions for Perl, Ruby, PHP, Python, C, Java and. NET". " O'Reilly Media, Inc.", 2007.

[72] JPcap, https://sourceforge.net/projects/jpcap , visited 16-05-2016

[73] DVWA, www.dvwa.co.uk , visited 03-01-2016

ملخص: Cross Site Scripting (XSS)هي مشكلة أمنية شائعة في تطبيقات الويب حيث يمكن للمهاجم حقن شفرة برمجية في مدخل التطبيق ثم يتم إرسالها إلى مستعرض الويب الخاص بالمستخدم. في مستعرض الويب يتم تنفيذ هذه التعليمات البرمجية وتستخدم لنقل البيانات الحساسة إلى طرف ثالث

. تحاول الحلول الحالية التخفيض من هجمات الـ XSS على كلتا جانبي الخادم والعميل، على سبيل المثال، بمراقبة وتعديل البيانات المرسلة من وإلى تطبيق ويب . يهدف حلنا للكشف عن هجمات الـ XSSعلى مستوى الـ Proxy بواسطة تحليل طلب الزبون واستجابة الخادم و هذا بالاختزال المشفر لكل كود سكريبت على صفحة الاستجابة لمقارنة هذه الاختزال المشفر مع اختزال السكريبتات الحميدة. إذا اكتشف النظام اي اختلاف في الاختزالين يتم حظر السكريبت.ثم تتم عملية اكتشاف نوع الهجوم من اجل حذف اي سكريبت خبيث تم حفظه في قاعدة البيانات.

مع هذه الطريقة نظامنا يقوم بحماية كل من الخادم والعميل .ولذلك، فإن المستخدم لديه طبقة إضافية من الحماية عند تصفح مواقع الانترنت.

الكلمات المفتاحية: web security, anti-XSS proxy, Cross-Site Scripting،XSS attack detection

Abstract: Cross Site Scripting (XSS) is a common security problem of web applications where an attacker can inject scripting code into the input of the application that is then sent to a user's web browser. In the web browser, this scripting code is executed and used to transfer sensitive data to a third party. Today's solutions attempt to prevent XSS on the server side and client side, for example, by inspecting and modifying the data sent to and from the web application. Our presented solution aims to detect XSS attacks on the proxy side by analyzing both the client request and the server response and hashing each found script on the response page to compare this hash with the benign one. If the system detects any content deviation, the script will be blocked, and the XSS type detector will be triggered to eliminate any stored XSS from database.

With such way our system does protect both server and client side. As a result, the user has an additional protection layer when surfing websites.

Keywords: XSS attacks detection, web security, anti-XSS proxy, Cross-Site Scripting.