International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

A Proposal: For Emerging Gaps in Finding Firm Solutions for Cross Site Scripting Attacks on Web Applications

Hima Bindu Sadashiva Reddy [a]

[a]*Atria Institute of Technology, Bangalore 560024, India*

DOI: *https://doi.org/10.55248/gengpi.2022.3.7.43*

ABSTRACT

There are many web vulnerabilities and popular among them is Cross Site Scripting Attacks (XSS). The XSS vulnerability can go to the extent of intruding on an organization's data via its web application. The activities the hijacker performs during these XSS attacks are accessing user sessions, deleting, adding, and modifying the data of the websites. Additionally, as they have control over the web pages, they add malicious code to distort the user interface and stop further business activities. If an organization's website is providing service across the globe, this would halt all the user transactions for many hours until the issue is resolved. The attackers would further proceed to access the organization's servers if the situation is not handled to stop the XSS attacks. These real-time scenarios explain the severity of the XSS attacks. Further implementing solutions to not face further attacks is still continuing. The reason for the quest to find better solutions is to avoid these XSS attacks, because the hackers are always finding various routes to hack these web applications. However, even after finding many solutions, these attacks are happening regularly. Hence it is necessary to discover the gap to find an appropriate solution even before any new XSS attack happens. This paper proposes a methodology to explore these gaps and solutions to an ongoing cross site scripting attacks.

Keywords:XSS, Web applications, web vulnerability, web attack, vulnerability detection, vulnerability exploitation

1. Introduction

Is everyone aware that the web applications you access are vulnerable to Cross Site Scripting (XSS) attacks?[1] Accessing web applications is an everyday normal activity these days, be it commenting on a food blog or an informational article, banking transactions, subscribing to a knowledge-based article, interactions on social networking sites, etc.[1]. All the activities performed to call for a dynamic interaction of the web application as user interaction is a required and frequent activity [1]. Yes, the web applications are vulnerable to Cross Site Scripting (XSS) attacks if the user's inputs are not validated properly [1].

Cross-Site Scripting attacks are causing panic in any industry that owns a web application. Even after running as many security scans as possible within different levels of the industry, including development, quality assurance, user acceptance testers, and IT security personnel. These attacks are still occurring. Where exactly are we failing to identify a possible firm solution? What precautionary measures are taken, and yet we are allowing hackers to attack the web application? Are there other factors that are causing the attacks? For example, is it specific to an industry or programming language? A real-time experience was in the year 2010 when an XSS attack occurred on the security company's (company's name intentionally kept confidential, it is one of the top five security companies) enterprise support web application. Solving the issue and getting the web application to be accessed globally was a nightmare for this company's IT professionals. Things didn't stop here; the team experienced these attacks almost very frequently (every three months for almost three years) [17]. The severity being enterprise support web application displaying a YouTube video and an animated gif image or distorting the original webpage.

Around 80% of the web applications in the entire World Wide Web are prone to Cross Site Scripting vulnerability [4]. This vulnerability is considered a serious and widespread security threat by the World Wide Web security [2, 4]. As this gives a cakewalk to the hackers to access user-

* *Corresponding author.* Tel.: +0-000-000-0000 ; fax: +0-000-000-0000.
E-mail address: author@institute.xxx

sensitive information. Researchers claim that the root cause for these vulnerabilities persist in many web applications due to developers' lack of experience. Because even after they have proposed multiple solutions to cross-site scripting attacks none have been followed and utilized efficiently [2]. According to the MITRE Common Weakness Enumeration Institute list of Top 25 Most Dangerous Software Errors, XSS ranks number one [2]. Recent results for topmost vulnerabilities according to the Open Web Application Security Project (OWASP), cross-site scripting attacks ranked second [2]. There are research studies proposing new solutions to developers and administrators by introducing security policies and certificates [3]. Introducing security-related policies and certificates is a strategy that would help to seamlessly integrate with generic web applications by conveying the SSL and secure redirect calls [3]. The major concern would be how sincere and dedicated will/have the personnel (developers and administrators) be/been in following these guidelines [15]. Highly useful policies would at least form a base in helping to overcome these severe attacks.

This research will be very helpful for people who are responsible for building and running a web application, World Wide Web users, and for web security's precautionary measures. The results generated from this research will give the complete picture of how the teams building and maintaining the web applications are well versed and aware of the serious impacts of the cross-site scripting vulnerabilities. The research intends to propose a method to identify the gap in making attack-free web applications from cross-site scripting. Additionally, explore various measures that are taken to avoid XSS attacks. Finally propose new solutions/precautionary measures and find a permanent solution.

2. Review of Literature

White Hat Security Company provides a detailed statistics report about web security every year [5, 13, 14,]. Observing the latest statistics report for year 2013, 2014 and 2015. Cross Site scripting ranked number one in 2013 and 2014, in 2015.Its ranking dropped to number three which is a good sign that industries are taking XSS vulnerability very seriously [5, 13, 14]. In year 2013 and 2014 programming languages like Java, Perl and PHP were the top three languages which were causing Cross site scripting attacks [5, 13, 14]. 2015 statistics report states that health care industry has the highest attack of XSS. However, 2013 and 2014 reports show gaming industry to be highest attacked[16].

xHunter tool is specifically designed to find which website URLs are prone to cross site scripting attacks; this tool would help web application owners to take precautionary measures [6]. One of the research study proposed defensive mechanism to avoid XSS attacks. As giant companies such as HSBC, Google search engine, Facebook, Vodafone etc. have come across these attacks mainly due to the lack of user input validations [7]. An algorithm was implemented by modifying the popular Firefox web browser to track the sensitive information during the XSS attacks. All these tests were conducted in a test environment, they even tried to protect the information leakage as well [8]. The only concern about this algorithm implementation is there is no evidence or citation of it being implemented in neither real time web applications nor the author's state.Although, Firefox Company has agreed to accept their solution. HTML and JavaScript form the major backbone in causing the XSS attacks [9]. There are many HTML and JavaScript tags the hackers come up with to attack web applications [10]. As cross site scripting attacks are easy to execute but difficult to prevent and trace, step by step detailed approach is proposed to avoid the attacks keeping in mind to not distort the user experience [11]. Detecting the flow of sensitive information is tracked based on the type of XSS attacks. The type of XSS attacks are Reflected Cross-Site Scripting attacks, Stored Cross-Site Scripting attacks, and DOM based Cross-Site Scripting attacks [11]. The proposed solution in this research is platform independent a well[11].

Developers are blamed for the attacks stating that the coding standards and guidelines are not followed. Is only development phase involved during software development life cycle? Why other phases of the software development life cycle such as design phase, code review phase, quality testing phase, and user acceptance testing phase not considered? Why there is no evidence stated as to why the development errors were not caught before the web application went live? On a different note, statistics report provides very useful information about the very serious vulnerabilities.Yet it does not state the reason how XSS vulnerability ranked number three in 2015 when compared to 2013 and 2014. With so many tools available to identify XSS vulnerability, none give details of how successful they in were making the web applications free from these vulnerabilities. Just providing tools to perform security scan to find these vulnerabilities is not sufficient.

3. Methodology

Design: Case study qualitative research tradition will be used in designing this qualitative study. For this case study users will be provided with Open-ended survey questions. Participants will be the developers, quality assurance testers, user acceptance testers, IT security specialists, project manager, project leads.

Sampling: Choosing industries which are affected highly based on the programming languages. To conduct a survey by collecting information from the department which faced this vulnerability. Category one (500 count) is to collect information from industry's department which have already had Cross Site Scripting attacks. Category two (500 count) is to collect information from industry's department with no Cross Site Scripting attacks till date.

Survey is conducted online and will take not more than ten to fifteen minutes [18]. Personal information collected will be kept confidential. Every department head will be personally contacted to obtain the details about the team members. Certain deadlines will be given to all the employees to send the survey answers. Reminders will be sent to both department heads and employees to complete the survey [19]. Anyone who has not submitted the survey will be contacted and requested to complete the survey.Explaining them the seriousness of the XSS vulnerability issue. Around six months will be spent in collecting the data.

These are certain questionnaires planned to be asked for the survey [20].Starting from design phase, planning phase will involve discussion about the security measures to be undertaken? [21]. Regarding development what standards are followed for security measures, in detail report of the security scan run just after development phase? [22].Was the security scan run for quality assurance testing, user acceptance testing, postproduction deployment? Reason why this attack(web application which had gone live) was not identified before it went live.Was it a new HTML injection please provide the complete details will the HTML injection tag? Were any bugs reported during any phase of software development life cycle? Or was it considered false positive provide reasons for the same? Was complete fix done for the reported bugs or XSS related vulnerabilities not done? Is it because either framework or design or user experiencebe interrupted? Please provide complete list of details about all the attacks from the HTML and JavaScript injection code? Which tools were used till date for scanning XSS vulnerabilities? Are the XSS scanning tools freely available? State the different security policies your team follows for developing a web application.What are the Different security certificates installed in your web application?

These questionnaires will help us learn how industries with no cross-site scripting attack have implemented the security measures related to XSS vulnerabilities. In addition to this we will learn how industries with reported XSS attacks have failed to maintain the security standards. It will give us insight on how the hackers are coming up with new techniques and code injection every time.

The survey or data collection is done through a web page [23]. The data submitted will be stored in PostgreSQL database. After the data is collected SQL query will be used to generate a report based on certain criteria. Consolidated report will be generated using Microsoft Excel.

Limitations: The accuracy of the information provided by the users is the first limitation identified.In an industry the same employee may not work for that team for longer period of time. If there are no proper documentations maintained regarding these vulnerability attacks, then new employees joining the team might not provide accurate information to this survey. Availability of the important participants is another limitation.For examplea participant was the main person involved in development of web application, and are on medical leave.

4. Summary

There are many research studies conducted so far. Every research provides details about the attack that have happened, and precautionary measurestaken and implemented.Yet we are still seeing the vulnerability exist after all the solutions, research and security policies provided. None of the authors have come up in explaining why this issue still exist even after proposing many solutions. There is no evidence as to why a permanent solution has not been discovered till date. At least a solution so that the similar attacks don't keep repeating. Consider for example an antivirus software we use in our PC's, it's very simple to understand.If we don't use this antivirus software our PC's will be attacked by deadly viruses which can damage our PC. That is the reason everyone who owns a PC's will definitely have antivirus software installed. Similarly developing software to bounce back the Cross Site Scripting attacks will be a very good solution to solve the XSS vulnerability problem. Developing this particular software is only possible only after finding out the possible gaps related to cross site scripting attacks. In order to find the gap, the proposed study should be implemented according to the details mentioned in the methodology section.

[1] Hydara, I, Bakar Md Sultan, A, Zulzalil, H, Admodisastro, N (2014). Current state of research on cross-site scripting (XSS) – A systematic literature review. Department of Software Engineering and Information System, 170 –186, 10.1016/j.infsof.2014.07.010

[2] Venkat Narayana Rao, T, Tejaswini, V, Preethi, K (2012) DEFENDING AGAINST WEB VULNERABILITIES AND CROSS-SITE SCRIPTING, Journal of Global Research in Computer Science, Volume 3, No. 5, 60 – 64

[3] Garcia-alfaro, J, Navarro-Arribas2, G. (2007). Prevention of Cross-Site Scripting Attacks on Current Web Applications. In Meersman, R.O.B.E.R.T. & Tari, Z.A.H.I.R (Eds), On the Move to Meaningful Internet Systems 2007: CoopIS, DOA, ODBASE, GADA, and IS (pp. 1770-1784).

[4] Dr. Shanmugam1, J, Dr. Ponnavaikko2, M (2008) Cross Site Scripting-Latest developments and solutions: A survey, Int. J. Open Problems Compt. Math., Vol. 1, No. 2, 102 - 121

[5] Whitehatseccom. (2013). Whitehatseccom. Retrieved 14 December, 2015, from https://www.whitehatsec.com/assets/WPstatsReport_052013.pdf

[6] Athanasopoulos, E , Krithinakis, E, Markatos, E, (2010) Hunting Cross-Site Scripting Attacks in the Network, Institute of Computer Science Foundation for Research and Technology, 1-8

[7] Amit Singh, A, Sathappan, S (2014) A Survey on XSS web-attack and Defense Mechanisms, International Journal of Advanced Research in Computer Science and Software Engineering Research Paper ,Volume 4, Issue 3 , 1160 – 1164

[8] Vogt, P, Nentwich, F, Jovanovic, N, Kirda, E, Kruegel, C and Vigna, G (2007) Cross-Site Scripting Prevention with Dynamic Data Tainting and Static Analysis, Secure Systems Lab Technical University Vienna, 1-12

[9] NSA, September 2011 Protect Against Cross Site Scripting (XSS) Attacks , Information Assurance Mission as National Security Agency, 1-2

[10] Kaur, G (2014) Study of Cross-Site Scripting Attacks and Their Countermeasures, International Journal of Computer Applications Technology and Research, Volume 3, Issue 10, 604 – 609.

[11] SHALINI1, S, USHA2, S (2011) Prevention Of Cross-Site Scripting Attacks (XSS) On Web Applications In The Client Side, International Journal of Computer Science Issues, Vol. 8, Issue 4, No 1, 650 - 654

[12] Practical Research: Planning and Design, P. D. Leedy & J. E. Ormrod, 11th (2015)

[13] Whitehatseccom. (2014). Whitehatseccom. Retrieved 14 December, 2015, from http://info.whitehatsec.com/rs/whitehatsecurity/images/statsreport2014-20140410.pdf

[14] Whitehatseccom. (2015). Whitehatseccom. Retrieved 14 December, 2015, from https://info.whitehatsec.com/rs/whitehatsecurity/images/2015-Stats-Report.pdf

[15] Rodriguez, K. M., Reddy, R. S., Barreiros, A. Q., & Zehtab, M. (2012, June). Optimizing Program Operations: Creating a Web-Based Application to Assign and Monitor Patient Outcomes, Educator Productivity and Service Reimbursement. In DIABETES (Vol. 61, pp. A631-A631). 1701 N BEAUREGARD ST, ALEXANDRIA, VA 22311-1717 USA: AMER DIABETES ASSOC.

[16] Kwon, D., Reddy, R., & Reis, I. M. (2021). ABCMETAapp: R shiny application for simulation-based estimation of mean and standard deviation for meta-analysis via approximate Bayesian computation. Research synthesis methods, 12(6), 842–848. https://doi.org/10.1002/jrsm.1505

[17] Reddy, H. B. S., Reddy, R. R. S., Jonnalagadda, R., Singh, P., & Gogineni, A. (2022). Usability Evaluation of an Unpopular Restaurant Recommender Web Application Zomato. Asian Journal of Research in Computer Science, 13(4), 12-33.

[18] Reddy, H. B. S., Reddy, R. R. S., Jonnalagadda, R., Singh, P., & Gogineni, A. (2022). Analysis of the Unexplored Security Issues Common to All Types of NoSQL Databases. Asian Journal of Research in Computer Science, 14(1), 1-12.

[19] Singh, P., Williams, K., Jonnalagadda, R., Gogineni, A., &; Reddy, R. R. (2022). International students: What's missing and what matters. Open Journal of Social Sciences, 10(02),

[20] Jonnalagadda, R., Singh, P., Gogineni, A., Reddy, R. R., & Reddy, H. B. (2022). Developing, implementing and evaluating training for online graduate teaching assistants based on Addie Model. Asian Journal of Education and Social Studies, 1-10.

[21] Sarmiento, J. M., Gogineni, A., Bernstein, J. N., Lee, C., Lineen, E. B., Pust, G. D., & Byers, P. M. (2020).Alcohol/illicit substance use in fatal motorcycle crashes. Journal of surgical research, 256, 243-250.

[22] Brown, M. E., Rizzuto, T., & Singh, P. (2019). Strategic compatibility, collaboration and collective impact for community change. Leadership & Organization Development Journal.

[23] Sprague-Jones, J., Singh, P., Rousseau, M., Counts, J., & Firman, C. (2020). The Protective Factors Survey: Establishing validity and reliability of a self-report measure of protective factors against child maltreatment. Children and Youth Services Review, 111, 104868

[24] Reddy Sadashiva Reddy, R., Reis, I. M., &Kwon, D. (2020). ABCMETAapp: R Shiny Application forSimulation-basedEstimation of Meanand Standard Deviationfor Meta-analysis via ApproximateBayesianComputation (ABC). arXiv e-prints, arXiv-2004.