Novel Approach for Web Threat Prevention against Vulnerability

Mr. S.A. Gadhiya [1],Dr. K.H. Wandra [2]
[1]*Research Scholar , C. U. Shah University, Gujarat*
[2]*Principal, Gujarat Maritime board Polytechnic, Gujarat*
Email: [1] sohil_2003it@yahoo.com,[2] khwandra@rediffmail.com

Abstract

Cross-Site Scripting is one of the major's attacks described by OWASP. The Cross Site Scripting attack is possible by inserting or changing the programming logic, changing and syntax of HTML elements by code injection attacks. The Web application is XSS Vulnerable when there is no proper input validation. The many web applications like social networking sites are the victims of this attack. This paper describes the various attack technique of cross site scripting and various mitigation involved in cross site scripting (XSS) Vulnerability. As new technology arrived, it comes with lot of new features and possibly attacks. In the today's trends of social web application, the SQL injection, cross site scripting (XSS) attack and cross site forgery attack are major challenges for web application.. The paper also describes various research perspective involved with cross site scripting. In the cyber world the security is now main issues for the user. The paper also shows demonstration of various cross site scripting (XSS) attack.

Key Words: Web Threats, XSS Attacks, XSS Prevention

1. Introduction

In the today's era of internet technology the users want everything should be on line. The increasing the use of internet and web application, it promotes E-commerce technology. Now the web users are crazy and they required everything on their hand. The Web developers and industries are also passionate to develop the web application. They use the edge cutting technology like cloud services, services oriented architecture [1], Web 2.0[1], Symantec web, Augmentation [2] to make people crazy. With increasing the use of internet, the technology gifts the term cyber crime and that's why now the security is major challenges for every web application. The paper describes various basics of XSS attack [3][4][5][6] and types. It also describe basic mitigation technique[3][4][6][7] involved with cross site scripting attack.. The paper divided into six sections. The section 2 describes various XSS attacks with example. The Section 3 describes the various mitigation technique involved with web application for XSS. The Section 4 describes the various research perspective involved with of XSS attacks.

2. Types of cross site Scripting attacks

There are mainly three types of XSS attacks [3][4][5] based on how the victim affected by attackers.

2.1. Persistence XSS

Persistence XSS [5][6]the malicious code is residing on server's data base and it is injection in victim's web pages when it received the web page of server. Basically the social web application is main target of such attacks.

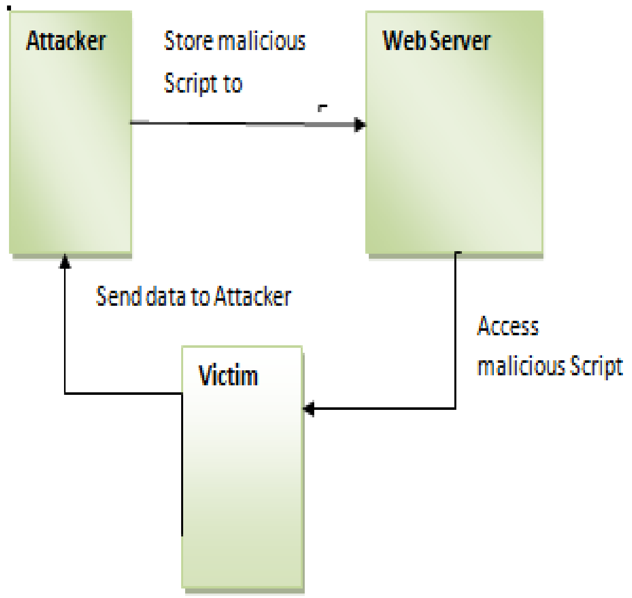


Fig. Persistence XSS

2.2. Non- Persistence XSS

In the non persistence attack[3][4][5], malicious code is part of the victim's request to the web application. The web application then includes such malicious string in the response sent back to the user.

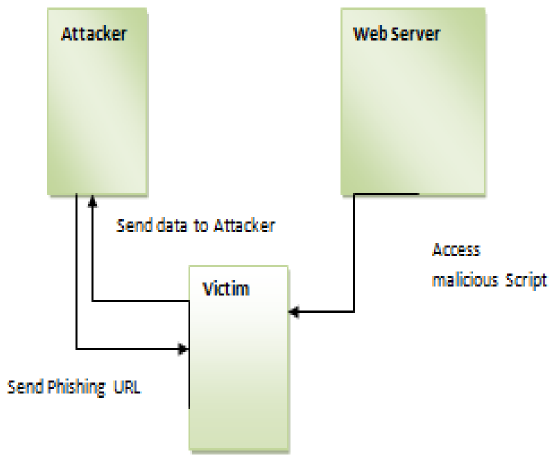


Figure – Non Persistence XSS

2.3. DOM based XSS

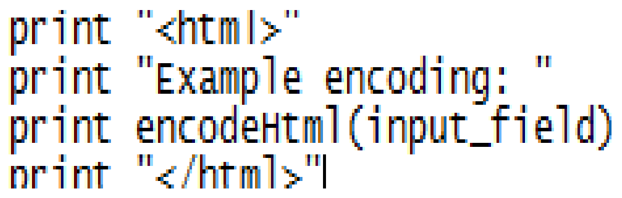
DOM-based XSS[3][4][5] is a combination of both persistent and reflected XSS. In DOM-based cross site scripting attack, first website's JavaScript is executed then the malicious code is parsed by the victim's browser using DOM.

3. The XSS Mitigation Technique

To the prevention of such type of code injection attack proper and secure input handling is required. There are two basic mechanisms for input handling.

3.1. Encoding/ Escaping Input

In encoding technique[4][6], every input data are escaped such that browser interprets every input as data. In the web development HTML escaping is most common type in encoding. Example it converts character like < into &lt;. We can perform encoding in client side and server side. The client side encoding is performed by client side script like javascript and vbscript. The server side encoding is performed by server side scripting language like PHA, ASP, JSP, ect.



```
print "<html>"
print "Example encoding: "
print encodeHtml(input_field)
print "</html>"
```

Fig- Example of escaping in PHP

3.1.1 Limitations of encoding

Using encoding it is not possible to encode each and every context of user input. With static data, it is not possible to use Dynamic HTML and DOM component in web pages.

3.2. Validation

Validation [4][6] is nothing but filtering mechanism to input data to remove malicious code. There are two main strategy of validation.

3.2.1 Blacklist

In Black-list[5], List of not valid elements is defined. Every element that does not match with black list are valid.

3.2.2 White list

In White-list [5], List of Valid elements are defined. Every element that is not part of White list is not valid.

4. Xss Research Prespective

In this section we discuss XSS Attack in research and literature perspective.

4.1 Omar Ismail,Masashi Etoh,Youki Kadobayashi, Suguru Yamaguchi, "A proposal and implementation of automatic detection/collection system for cross-site scripting vulnerability" 2004 [8]

In this paper, authors propose a client-side system that automatically detects XSS vulnerability by manipulating either request or server response. The system also shares the indication of vulnerability via a central repository. The purpose of the proposed system is twofold: to protect users from XSS attacks, and to warn the web servers with XSS vulnerabilities

4.2 Abdul Razzaq, Ali Hur, Nasir Haider, Farooq Ahmad, "Multi-Layered Defense against Web Application Attacks", 2009 [9]

In this paper, authors have introduced a novel approach of multiple layered defenses to the application level attacks which possess higher detection ability with low false positive rate. The System is capable to detect application level known and unknown attacks especially XSS and SQL injection, in efficient way. This system is also helpful for the developers to find the application vulnerabilities well in time by visually observing the proper validation through validation flow graph.

4.3 Adam Kie zun, Philip J. Guo, Karthick Jayaraman, Michael D. Ernst ,"Automatic Creation of SQL Injection and Cross-Site Scripting Attacks", 2009

In this paper, the authors present a technique for finding security vulnerabilities in Web applications. SQL Injection (SQLI) and cross site scripting (XSS) attacks are widespread forms of attack in which the attacker crafts the input to the application to access or modify user data and execute malicious code. The authors also present an automatic technique for creating inputs that expose SQLI and XSS vulnerabilities.

4.4 Rattipong Putthacharoen, Pratheep Bunyatnoparat, "Protecting Cookies from Cross Site Script Attacks Using Dynamic Cookies Rewriting Technique", 2011 [11]

In this paper, authors introduce a new technique called “Dynamic Cookies Rewriting”, this technique aims to render the cookies useless for XSS attacks. The technique is implemented in a web proxy where it will automatically rewrite the cookies that are sent back and forth between the users and the web applications. With our technique in place, the cookies at the browser’s database now are not valid for the web applications; therefore the XSS attack will not be able to impersonate the users using stolen cookies.

4.5 Yi Wang, Zhoujun Li,Tao Guo , “Program Slicing Stored XSS Bugs in Web Application”, 2011 [12]

The authors uses Stored XSS detection algorithm to locate possible vulnerabilities and then uses program slicing method to generate the program slices related with the detected vulnerabilities. In contrast to traditional static tainting-based approaches for detecting Reflected XSS, authors adapt the algorithm for detecting Stored XSS and generate the complete related program slices further for later manual checking or dynamic analysis.

4.6 Hossain Shahriar,Mohammad Zulkernine "S2XS2: A Server Side Approach to Automatically Detect XSS Attacks", 2011 [13]

In this paper, the authors develop an automated framework to detect XSS attacks at the server side based on the notion of boundary injection and policy generation. Boundaries mark content generation locations in server script code. The authors develop a prototype tool to automatically insert boundaries and generate policies for JSP programs. The authors evaluate the approach with four JSP programs. The results indicate that the approach detects most of the well known XSS attacks.

4.7 HossainShahriar,Mohammad Zulkernine, "Injecting Comments to Detect JavaScript Code Injection Attacks", 2011 [14]

In this paper , the authors develop a server side approach that distinguishes injected JavaScript code from legitimate JavaScript code. The approach is based on the concept of injecting comment statements containing random tokens and features of legitimate JavaScript code. When a response page is generated, JavaScript code without or incorrect comment is considered as injected code. Moreover, the valid comments are checked for duplicity. Any presence of duplicate comments or a mismatch between expected code features and actually observed features represents JavaScript code as injected.

4.8 L.K. Shar, H.B.K. Tan , “Auditing the XSS defence features implemented in web application programs”, 2012 [15]

In this paper, On the basis of the possible implementation patterns of defensive coding methods, authors approach a defenses implemented for securing each potentially vulnerable HTML output. It then introduces a variant of control flow graph, called tainted information flow graph, as a model to audit the adequacy of XSS defense. The authors evaluated the proposed method based on the experiments on seven Java-based web applications.

5. Conclusion

As a Cross site scripting attack is very dangerous for web application. The web application must have one or more prevention mechanism to get protected from XSS attack. However with the progress of technology, a new approach of attack are invented and possibly prevention are also discovered. So, to prevent a web application form XSS attack is very difficult task and the protection mechanism must updated for latest possible XSS attack.

6. References

[1] Adnan Masood,"Cyber Security for Service Oriented Architectures in a Web 2.0 World: An Overview of SOA Vulnerabilities in Financial Services", HST-2013, pp 1-6.

[2] S. A. Gadhiya, K. H. Wandra, V. B. Vaghela ," Role of mobile augmentation in mobile application development",AICERA-2011, pp 1-5.

[3] Vikas K. Malviya, Saket Saurav,Atul Gupta, "On Security Issues in Web Applications through Cross Site Scripting (XSS)", APSEC-2013,pp 583-588.

[4] https://www.owasp.org/index.php/Top_10_2013-A3-Cross-Site_Scripting_(XSS)

[5] http://excess-xss.com/#xss-attacks

[6] http://excess-xss.com/#xss-prevention

[7] Rahul Johari, Pankaj Sharma, "A Survey On Web Application Vulnerabilities(SQLIA,XSS)Exploitation and Security Engine for SQL Injection " , CSNT-2012, pp 453 -458.

[8] Omar Ismail Masashi Etoh Youki Kadobayashi, "A Proposal and Implementation of Automatic Detection/Collection System for Cross-Site Scripting Vulnerability", AINA-2004, pp 145-151.

[9] Abdul Razzaq, Ali Hur, Nasir Haider, Farooq Ahmad,"Multi-Layered Defense against Web Application Attacks", ITNG-2009, pp 492-497.

[10] Adam Kie˙zun, Philip J. Guo, Karthick Jayaraman, Michael D. Ernst, "Automatic Creation of SQL Injection and Cross-Site Scripting Attacks", ICSE-2009, pp 199-209.

[11] Rattipong Putthacharoen, Pratheep Bunyatnoparat,"Protecting Cookies from Cross Site Script Attacks Using Dynamic Cookies Rewriting Technique", ICACT-2011, pp 1090-1094.

[12] Yi Wang, Zhoujun Li, Tao Guo "Program Slicing Stored XSS Bugs in Web Application", TASE-2011, pp 191-194.

[13] Hossain Shahriar, Mohammad Zulkernine, "S2XS2: A Server Side Approach to Automatically Detect XSS Attacks", DASC-2011, pp 7-14.

[14] Hossain Shahriar, Mohammad Zulkernine"Injecting Comments to Detect JavaScript Code Injection Attacks",COMPSACW-2011, pp 104-109.

[15] L.K. Shar,H.B.K. Tan,"Auditing the XSS defence features implemented in web application programs",ITE-2012, pp 377 - 390.