International Journal of Engineering Research-Online
A Peer Reviewed International Journal
Articles available online http://www.ijoer.in

Vol.3., Issue.3, 2015

REVIEW ARTICLE

ISSN: 2321-7758

MEASURES TO PREVENT WEB ATTACKS

MANI SHARMA
Department of Computer Science & Engineering
Mahatma Gandhi Mission's Engineering College, Noida, India

Article Received: 29/04/2015 | Article Revised on:03/05/2015 | Article Accepted on:06/05/2015

ABSTRACT
In this world of changing technologies, websites are facing serious problems of information leakage due to code injection attacks. Attackers hack the websites for the purpose of information stealing of users. Various investigations are performed by the researchers on web attacks and they found that XSS is the topmost among various attacks faced by popular sites. Various techniques developed by the researchers to protect websites from information leakage but they suffers the weakness of complex construction, manual work requirement, performance overhead etc. In this paper , we bring solution how to secure websites against these attacks. This research paper shows a new approach which overcomes the limitation of previous techniques .This research paper suggest some improvements to prevent web attacks. We believe that approach and measures suggest by us will help in securing the websites from code injection attacks in future.

INTRODUCTION

Due to regular change in technologies, websites are using various client side scripting languages such as JavaScript, VBScript, ActiveX to increase their usability. Hence the use of these client side languages leads to serious vulnerabilities faced by websites. XSS, SQL Injection, XSRF are some of the attacks which are mostly faced by websites.XSS is the topmost among various code injection attacks faced by websites. HSBC, Google Search Engine, My Space, Vodafone are some popular websites which have faced XSS attack.

XSS is code injection attack in which the hacker injects the malicious JavaScript in the output of the website. When user visits this webpage, script gets downloaded and executed .When this script gets executed all the confidential information of the user is transferred to the hacker.XSS is difficult to detect and prevent as compared to other code injection attacks.

Content Security Policy (CSP) is the solution of these content injection attacks. It was developed by Mozilla Foundation and implemented first in Firefox4. It is a mechanism that identifies and prevents XSS and other code injection attacks. So, many popular websites uses this policy to protect themselves from these attacks.

This research paper suggests the measures and solution of these code injection attacks. This paper suggests certain improvements which not only prevent XSS but also other code injection attacks.

This research paper is divided into three sections. First section describes the literature

survey. Second section describes concept of our approach. Third section describes suggested improvements. Rest of the paper describes conclusion and future work.

SECTION-1

LITRATURE SURVEY

Existing techniques suffers the limitations of complex construction of device, runtime overhead, manual work requirement etc. In case of [2] SWAP, there is performance overhead as each page has to be passed through the JavaScript tester before it reaches to the client. Hence, it will take more time for the data to reach to its destination. Second limitation is that JavaScript tester will not able to identify the malicious content of other scripts such as ActiveX, VBScript etc. Third, it is a server side solution and depends upon the service providers. When the service providers are unwilling or incapable to provide security to content of websites, they left the users defenceless.

In the case of [1] Noxes, first developed client side solution, does not provide the procedure to identify errors. Second it needs watchful configuration as when the connection rules are mismatched then it prompts the user either to allow or block the connection.

Although Noxes provides complete functionality but it lags behind in some work:

1. Plans are making by researchers to make availability of tool as free utility.
2. Noxes currently lags SSL support.

Researchers investigated that [3]CSP is facing challenges and lagging behind than other policies because of the following reasons:

1. Removing of inline scripts leads more time for the user to upload the webpage. Hence, there is delay in reaching data to its destination.
2. Use of CSP results in risk of breaking functionalities because as high as a website is secured less it would be involve in business activities.

Hence, we analyzed that no technique is completely efficient to provide security to website as they suffers from certain limitations.

In this paper, we focused on the problems of content injection attacks. We are providing a solution by introducing our approach and suggests measures to prevent code injection attacks.

SECTION-2

APPROACH

Our approach is based on the following steps:

1. Copy the content: Data which is to be transmitted from sender to receiver is copied for protection.

2. **Install a firewall**: A firewall is installed to detect and filter the malicious content of website.

3. **Selection**: Commands are used to disable keyboard. Only mouse is used for the purpose of scroll down.

SECTION-3

SUGGESTED IMPROVEMENTS

1. Websites must introduce some new standards in order to improve its efficiency and performance. New standards will help in providing security measures for these code injection attacks. These standards will identify and not allow these attacks to be execute.
2. Tools must be created for web browsers which can raise warning or stop submitting SQL commands directly from browsers.
3. Web application servers can have configuration settings that will not allow to submit SQL queries directly from browser clients.
4. Search Engines like Google and browser should have some kind of procedure through which they can identify websites which are vulnerable and blacklist them.
5. Users should make aware to prefer for only SSL based sites for any monetary transactions.

We believe that improvements suggested by us will help to alert all the people among servers and users. These improvements will secure the content of website in future.

CONCLUSION AND FUTURE WORK

We have discussed the problems of web attacks. Then we have mentioned the limitations of existing techniques as “what were the reasons of their failures”? As it not possible to stop these type of attacks completely with existing solutions. We have mentioned our approach and its working. We have suggested some improvements to detect and prevent XSS, SQL Injection and other web attacks. Finally we conclude that awareness must be created among all people to secure data of websites. We hope that approach and improvements suggested by us will help to prevent information leakage in websites.

ACKNOWLEDGEMENT

We are happy to express our thanks to Mohammad Asim and all other teachers for their guidance. We are also like to thanks to our colleagues and all those people who have helped directly or indirectly to complete this research paper.

REFERENCES

[1]. Engin Kirda, Christopher Kruegel, Giovanni Vigna, and Nenad Jovanovic. Noxes: A client-side solution for mitigating cross site scripting attacks. In Proceedings of the 21st ACM Symposium on Applied Computing (SAC), 2006

[2]. WURZINGER , PLATER, LUDL , KIRDA , KRUEGAL: SWAP : mitigating XSS attacks using reverse proxy.