Cross site scripting attack

Shweta A. Lakhapati[1], Prof. P. V. Shirbhate[2], Shivani Jagtap[3], Ashwini Shrirang[4]

Abstract **— Web application (WA) enlarge its usages to provide more and more services and it has become one of the most essential communication channels between service providers and the users. Most of the web applications have security vulnerabilities like XSS (Cross Site Scripting) attacks, phishing attacks which are exploited by the attackers to hack the credential and personal data from the web application for malicious purpose. To augment the users' experience many web applications are using client side scripting languages such as JavaScript but this growing of JavaScript is increasing serious security vulnerabilities in web application too, such as cross-site scripting (XSS).**

Key Words **— security vulnerabilities, credential, , phishing, JavaScript.**

I. INTRODUCTION

In this modern world, web application (WA) enlarge its usages to provide more and more services and it has become one of the most important communication channels between service providers and the users. To build up the users' experience many web applications are using client side scripting languages such a JavaScript but this rapid growing of JavaScript is increasing serious security vulnerabilities in web application too. XSS is a computer security vulnerability mostly found in the web appliaction . XSS vulnerability is among the top web application vulnerability according to OWASP. OWASP is an institute. In cross site scripting attack an attacker attacks on the vulnerable site and exploit the whole content in the website.

In the cross site scripting attack, the attacker inject bad script into the user server and when the user request to the user server, user receives the bad script from user server due to this user goes under the attacker's control and in that case the attacker steal the valuable data from user. Cross Site Scripting (XSS) is differ from SQL injection in the way that XSS targets the client"s browser i:e the victim. Cross site scripting occurs on the client side. In this attack, attacker inject the malicious code in the form of any programming language such javascript.

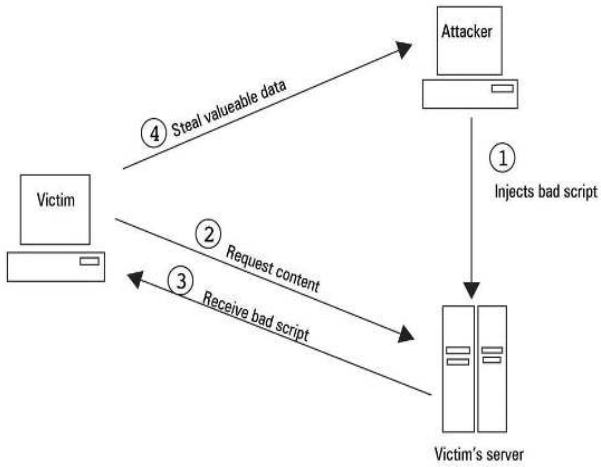

Fig 1: Operation of XSS

II. WHAT IS CROSS SITE SCRIPTING ATTACK?

Cross-site scripting (XSS) is a type of computer security vulnerbility typically found in web applications. XSS enables attackers to inject client-side scripts into web pages viewed by other users. Cross site scripting attack is code injection that the hacker inject a malicious script into the other users browser. In cross site scripting attack, attacker search the vulnerable website for the injection of malicious script into the browser.

- Cross site scripting attack happens to sniff the users documentations.

- Cross site scripting attack happens to divert the some other pages.
- Cross site scripting attack happens to pop up the data.
- Cross site scripting attack happens to do business loss of user.

The malicious java script code is executed by the web browser as a typical code. Browser cannot identify the typical code and injected malicious code. Generally Cross Site Scripting attacks can be classified into three categories.

- Reflected XSS (Non Persistent)
- Stored XSS (Persistent)
- DOM (Document Object Model)

III. TYPES OF CROSS SITE SCRIPTING

There are three distinct types of XSS attacks:

1. **Reflected XSS:**

Non-persistent cross-site scripting vulnerability is also known as non persistent xss. It is the most common type found now a days. The attack code is not persistently stored on the server, but, instead, it is immediately reflected back to the victim via HTTP request.

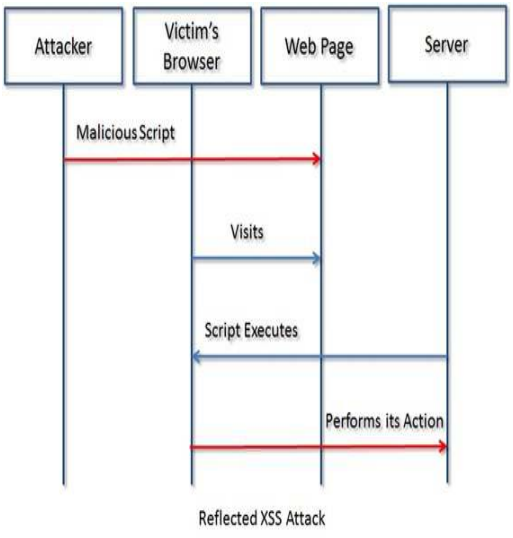

Fig 2: Reflected XSS Attack

The server insert the input with the html file and return the file to browser. In this type of the attack script is not permanently store on the server but this script reflected back to user.

In this type, the attacker inject a malicious code on the web page, and when the victims visit that page server execute this script on the victims browser and then victims browser performs its action on the server.This kind of xss vulnerability frequently occur in search field. In this manner the reflected XSS happens.

For example:

Let us consider a project hosting website. To find our favourite project,we will just input the related work in the search box. When searching is finished, it will display a message like this "search result for your word". If the server fail to sanitize the input properly, it will result in execution of injected script. In case of reflected xss attack, attacker will send the specially crafted link to victim and trick them into click the link. When user click the link, the browser will send the injected code to server, the server reflect the attack back to the users browser. The browser the execute the code.

2. ***Persistent XSS:***

A malicious script is injected in web application and is permanently stored on the server.Hence those attack is known as stored XSS attack.

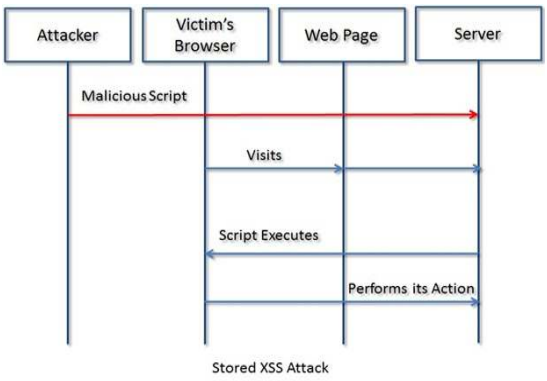

Fig 3: Stored XSS Attack

In this type of attack, the attacker inject malicious javascript directly on the server and when the victim browser

visit the web page which contain malicious script then the server execute this script on the victims browser and at end victims browser perform its action on the server.

For example:

Many website host a support forum where registered user can ask their doubts by posting message, which are stored in the database. Let us consider, an hacker insert a message containing malicious script. If the server fail to sanitize the input provided, it result in execution of injected script. The code will be executed whenever a user try to read the post.if suppose the injected code is cookie stealing code, then it will steal cookies of user who read the post. Using the cookies,attacker can take control of your account.

3. *DOM-based:*

Dom based cross site scripting is also knows as server side attack. A Dom-based XSS attack is triggered on the client side [19]. DOM allows dynamic scripts, including JavaScript, to reference the document's components – for example, a session cookie or a form field. Such susceptibility could be triggered when an active content (for example, a JavaScript function) is altered by a specially created request, allowing a DOM element to be manipulated by an attacker. This attack is in contrast to other xss attack, wherein the attack payload is placed in the response page.

IV. PREVENTION METHOD OF XSS

Preventions of XSS is very important. There three methods of preventing XSS. These are as follows:

- **liberating:**

liberating means to remove the malicious script which is added by the attacker into the website from the website database. The attacker inject the SQL injection into the database which contain the all information about the website.proper coding which escape the user input is as a data not as a code. The input which is in the messege box should contain HTML tags not only html.

- **Authenticating input:**

Authenticating input means valid input should be inserted into the messege. The user input should be authenticated before its use by the web application.the application state should be maintain unconstraint. The attacker firstly target the database which contain the whole information about the website. Authentication means providing security because third party shouldn't interfair in that.

- **Sanitizing:**

Sanitizing has done in two ways i.e inbound sanitizing and outbound sanitizing. The developer insert a special character while developing the website that time this special character doesn't sanitize at the backend then because lack of sanitization vulnerability occure in the website. So proper sanitization must need. Sanitizing

V. EXISTING METHOD:

A. Static analysis:

A static string arbiter checks the string output of a program with context free grammar. This technique checks the presence of "<script>" tag in the whole document. The web application vulnerability is detected by a static analysis tool pixy. Non persistent attack is detected by pairing the incoming data and outgoing java script using a simple metric like matching the incoming data to HTML java script code. The static analysis is used to find out all inline java script code in web pages. In Browser enforced embedded policy, modified the web application and embedded some policies. Policy exits a hook function that will run before execution of any script. The web request parameter passed to the HTML parser. They modified the HTML, JAVA script tags, method, method calls and expression along tokens.

B. ***Dynamic analysis****:*

There are two approaches in the dynamic analysis, vulnerability analysis based approach and attack prevention approach. Vulnerability analysis means to check the untrusted data and remove this untrusted data from the content. There are two aproches in this, interpreter based approach and grammatical structure analysis. Pietraszek, and Berghe are two scientist use approach of remaking interpreter to track untrusted data at the character level and to identify vulnerabilities.

A successful inject attack changes the grammatical structure of the exploited entity. Augment the user input with metadata to track this sub-string from source to sinks. Only checking the syntactic structure is not sufficient to prevent this sort of workflow vulnerabilities that are caused by the interaction of multiple modules.

In attack prevention approach there are two approaches for the detection of XSS attack.proxy based solution and browser enforced embedded policies these are two approaches. a web proxy protects against transferring of sensitive information from victim's site to third party's site. proxy-based solution doesn't present any procedure to identify the errors and it needs watchful configuration.

VI. Major constituents that cause the XSS vulnerabilities

There are two major constituents that cause the XSS vulnerability:

- **Sructural requirements needed for XSS attack:**

Attacker have developed many tools to target the web sites which are available for easy in internet. The most basic data managment for these vulnerabilities are very simple to perform. The entry points of the vulnerable XSS web applications can be found using direct tools inclusive of Google. Only web browser is needed to attack a web application. No proper tools are need for hacking the web application. The attacker target the database directly which contain whole data about website.

- **Community factors:**

Web applications are developed by developer with varied experience and with a very little knowledge of security aspect of an application. Further web application comprises of one or more web pages that are developed by a group. Hence the chances are very high that though rare web pages are structure with security mechanisms where in other few could be leave. XSS vulnerabilities arise due to coding issues. The coding vulnerabilities transfer from site to site and there is no single patch available to fix all the XSS vulnerabilities.

VII CONCLUSION

This is my analysis paper on most well-known injection issues, cross-site scripting. The information contained in this paper could be very useful for new application/web developers for developing smarter and secure applications running over the web. Web Application performs many critical tasks and deals with sensitive information. Nowadays, web application facing security problem for these injection problem and XSS is one of them. This paper proposes a novel, client-side solution to this problem.so proper coding must done.

REFERENCES

[1]Suman Saha, Consideration Points: Detecting Cross-Site Scripting Dept. of Computer Science and Engineering Hanyang University Ansan, South Korea.

[2] Jayamsakthi Shanmugam1, Dr. M. Ponnavaikko2, Cross Site Scripting-Latest developments and solutions: A survey, 1Research Student, BITS, Pilani, India, 2Vice Chancellor, Bharathidasan University, India.

[3] Cross-Site Scripting Prevention with Dynamic Data Tainting and Static Analysis: Philipp Vogt§, Florian Nentwich§, Nenad Jovanovic§, Engin Kirda§, Christopher Kruegel§, and Giovanni Vigna‡

AUTHOR'S PROFILE

Passport Size Latest Color	**Shweta A. Lakhapati** BE final year, JDIET, yavatmal Email id: shweta.lakhapati01@gmail.com

Passport Size Latest Color Photo	**Prof. P. V. Shirbhate** Assistant Prof. IT department Email id: priyavshirbhate@gmail.com

Passport Size Latest Color Photo	**Ashwini Shrirang** BE third year, JDIET,yavatmal

Passport Size Latest Color Photo	**Shivani Jagtap** BE third year, JDIET,yavatmal