Volume 5, Issue 5, MAY 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Detection and Prevention of Javascript Vulnerability in Social Media

V. M. Vasava, Prof. Rupali A. Mangrule
CSE Department, MIT, Aurangabad,
Maharashtra, India

Abstract: ***Now days, everybody is utilizing Internet as an asset for picking up data. So, this reason individuals experience numerous quantities of sites however out of 10 sites more or less 7 to 8 sites may be powerless from the security perspectives. There are different varieties of JavaScript security issues like improper client-server trust relationships, exposure in browser plug-in code and incorrect implementation of security policy that can increase the risk for users.***

The only way for organizations to avoid these JavaScript security risks is to develop and source applications that are free of JavaScript security vulnerabilities. The different organization use JavaScript security analyzers to test for and remedies these vulnerabilities.

Static tools like Imposter, spectator, JSHint, AMO validator etc. exist, to check the code of web pages for vulnerability and they use the methods like regular expression, pattern matching etc. but none of them is 100% reliable. So, the proposed strategy is to centering recognizing and preventing the JavaScript Vulnerability like, phishing attacks, Clickjacking based on server side approaches.

Furthermore by executing the proposed undertaking can conclude that phishing attack detection and prevention with an increased accuracy rate. Also it additionally diminishes the achievement rate of attack on Clickjacking attacks in social networking sites.

Keyword: Vulnerability, Clickjacking, X-frame-options, Error Rate, Phishing, Vulnerability Analysis

I. INTRODUCTION

Now days, everyone are using social media sites for to gather in detailed personal and professional information, content sharing, interaction between users [2]. With the adventure of online social media like Facebook, LinkedIn, Google+ ,Twitter, Amazon, eBay, PayPal *etc*. the web based attacks like Phishing, Clickjacking, cookie stealing has rapidly increased. Vulnerability is a weakness in system which allows attackers to reduce the system performance, assurance and security [1].

Vulnerability analysis is a process of assessing the security of an application through auditing of either the applications code or behavior for possible security problems. It can be done in two ways –static and dynamic analysis.

Phishing is a type of attack in which e-mail messages designed to look like messages from a trusted agent, such as a bank, auction site, or e-commerce site that used to steal personal and financial data of victim. The victim personal information and sensitive is stolen through a dummy webpage by attacker, where in people believe that they are dealing with an authorized party, like their bank [5].

Clickjacking is a web based attack that first introduced by Jeremiah Grossman and Robert Hanson in 2008 during their research on web application security. It is mainly a browser security issue that allows malicious scripts to be executed on the client side and to carry out Clickjacking attacks in on all web browser platforms [12].

ClickJacking or User Interface redress attack is a technique to trick the web user into clicking on something different from what the user might expect. This can be used to perform an action with user's credentials.

II. LITERATURE SURVEY

In a market, there are various mechanisms of detection and prevention for phishing attack, clickjacking attack. Many security researchers provided solutions against those techniques related web based attacks but not guarantee to provide solutions and still around vulnerable applications.

A. Phishing Attack Detection Mechanism

The security researchers Anupama *etc.* have investigation the different kinds of solutions against phishing attacks and also developed a phishing tools (PhishAri) for real-time detection of phishing URL at twitter. The tools provide three mechanism used to detection for real time phishing attack such as 1.URL based feature 2.WHOIs based 3.Network Based. It should be extract the feature and check the whether the URL phish or not such as twitter features like tweet content, length, hash tags *etc*. These tools were use machine learning classification technique and detect phishing tweets with an accuracy of 92.52% [22]. According to Sunil Chaudhary, "Recognition of Phishing Attack s utilizing anomalies

in phishing website" have surveyed experimentally contrasting association classification algorithms, i.e. Classification Based Association (CBA), and Multi-class Classification based on Association classification with other traditional classification algorithms and recognizing the web page phish or not [9].

The website URL and text feature based approach (*e.g.*, Huang et al. 2012) focuses on the characteristics of the URL and text content of a target website. A content based approach to detecting phishing websites called CANTINA. The CANTINA (Zhang et al. 2007) examines the content of a web page to determine whether it is legitimate or not.

The drawback of this all approach is limited feature and it can't be strong security against phishing attacks.

These detection and prevention methods of phishing attacks are as follows.

1. URL Features
2. Listing method
3. Email Based approach
4. Heuristics Based Approach

i) URL Features

The general features of phishing detection criteria [18] as below.

1. Web Address Bar
- Long URL Address
- Replacing Similar Characters for URL
- Adding Prefix or Suffix
- Using the @ Symbol to Confuse
- Using Hexadecimal Character Codes

2. URL & Domain Identity
- Using IP Address
- Request URL
- URL of Anchor
- DNS Record
- Abnormal URL

3. Security & Encryption
- SSL Certificate
- Certification Authority
- Abnormal Cookie
- Distinguished Names Certificate (DN)

4. Source Code & Java Script
- Redirect Pages
- Straddling Attack
- Pharming Attack
- Using on Mouse Over to hide the link

5. Page Style & Contents
- Spelling Errors
- Copying Website
- Using forms with "Submit" button
- Using Pop-Ups Windows
- Disabling Right Click

6. Social Human Factor
- Much emphasis on security & response
- Public generic salutation
- Buying time to access Accounts

ii) Listing Method

List based methods classify websites into either phishing or trusted one and maintain into database lookup in the form of either black list or white list. These lists can be of IP addresses or domain name or URL's. Black list is list of IP Addresses or domain name of phishing websites. While white lists is a list of IP Address or Domain name or URLs collection of legitimate websites [8].

B) Clickjacking Attack Detection Mechanism

OWASP defined Clickjacking which is also known as a "UI redress attack" is performed when the attacker uses multiple transparent or opaque layers to trick a user into clicking on a button or link on another page when they were intending to click on the top level page. Thus, the attacker is hijacking the clicks and redirect to another page, most likely owned by another application, domain or both. Using a similar technique, keystrokes can also be hijacked [16].

- **Clickjacking Methodology**

The attackers are loading another page over it in a transparent layer on clickjacked page. The clients feel that they are clicking visible buttons or visible GUI, while they are actually performing actions on the hidden page. For example, the

hidden page may be an authentic page, in this way the attackers can trap clients into performing activities which the clients never expected and there is no way of tracing such actions to the attackers.

- **Exploitation**

The Basic ingredients to prepare for a clickjacking attack are:

Iframe – This is a frame in HTML that frames a webpage in it.
Z-index – decides the iframe index in the stack.
Opacity – makes the iframe transparent.
Position: Absolute – lines up the iframe with the dummy page.

There are different types of Clickjacking attacks as follows.

1. Likejacking
2. Cursorjacking
3. Click jacking and CSRF
4. Click jacking and XSS
5. Strokejacking

III. PROPOSED SOLUTIONS

The proposed technique is to demonstrate the phishing attack as given steps.

1. First of all you need to create a fake login page of Gmail.
2. Then Go to fake Gmail webpage and enter username and password in a Gmail account.
3. Now, establishing connection through SMTP protocol get online Gmail user account verified & successfully login.
3. Now, open that fake login page and get its URL from the browser. Now, everything is done by attacker. You can send this credential information directly to attacker through log files.
4. To spoof an URL, there are many ways but here its represent the simplest method to capture credential information.

A) Defense Techniques

1. Phishing Attack

The proposed strategy is to prevention of the phishing attacks in view of URL Features and Listing Method.

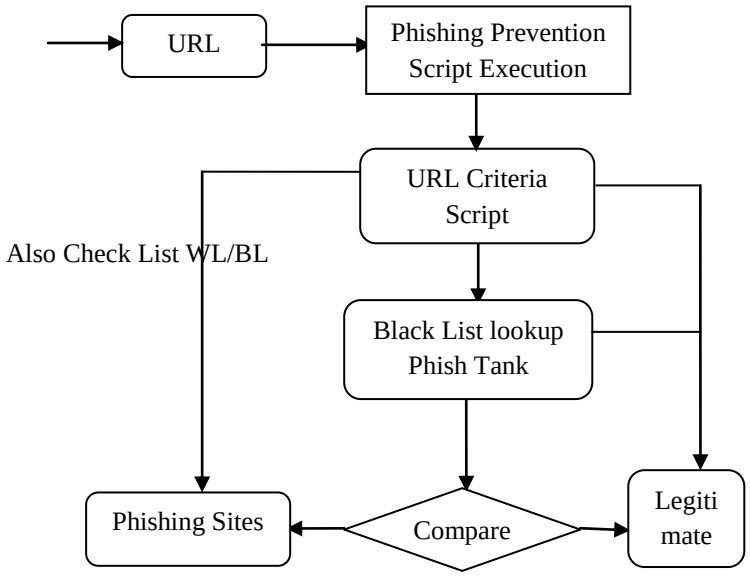

Fig. 1: Workflow of Phishing Prevention Diagram

The proposed strategy is used to combine features for prevention of phishing attacks. These Features are as below.

1. URL Features

IP Address in URL
The number of '.' in URL
The number of slash (/) in URL
No. of suspicious URL (‘@’,’-‘,’_”)
Long URL
Adding Prefix or Suffix
Using Hexadecimal Character Codes

2. Black List /White List Features

Black List
White List

The rule base algorithm has input parameters (criterion) and one output that contain all the “IF-THEN” rules of the system.

2. Clickjacking Prevention

The proposed algorithm for prevention of Clickjacking attack is as follows.

INPUT: HTML Web page source code

OUTPUT: Warn Message pop-up dialogue if malicious detected or not

1. Find all javascript code and assign elems variable in web page
2. Search Opacity tag value
3. If (Found iframe with opacity 0)
   Alert ("You are clicking hidden button or fake button")
4. Else
   Alert ("Visible in iframe")
5. Block Page using Html Tag
6. Exit

The process flow chart for clickjacking prevention as follows.

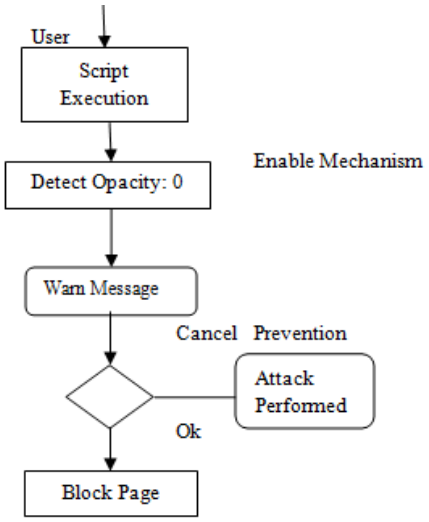

Fig. 2: Prevention of Clickjacking

IV. RESULT EVALUATION

A. Phishing Attack

1 Performance Measurement

The precision of the forecast model is assessed utilizing the URL set of criteria.From the results, it is clear that the measure of accuracy using random classification algorithem based proposed criteria is better as compared to existing criteria.It improves the accuracy & max criteria to be set for achieving higher detection error rate in phishing url [9].The following formulas are used to calculate performance measures of phishing prevention algorithm use both URL Features set and Listing method given current phishing URL data.

Table 1: Given Phishing URL Features Sets data

Criteria	F1	F2	F3	F4	F5	F6	F7
	0	0	0	0	0	1	0

Exisitng Criteria :

Average<%> Error rate=(Feature set or Listing Method)/Total no of Features

Proposed Criteria :

Average<%> Error rate=(Feature set +Listing Method)/Total no of Features

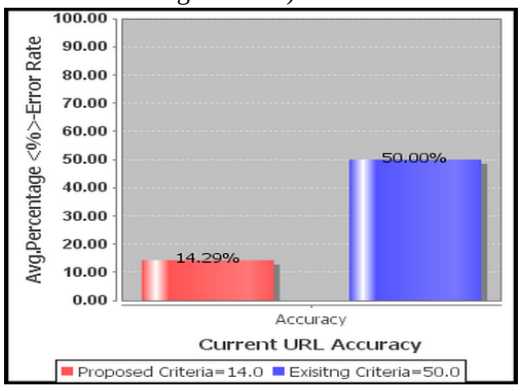

Fig. 3: Phishing Prediction Accuracy Comparison

2 Error Rate Detection Performance Analysis

Next, the proposed approach is investigating the feasibility of real-time application. In order to prevent Internet users from clicking on phishing URLs in real-time [4], such a proposed system needs to be highly accurate and needs to have tolerable low response time.

Table 2: Phishing Prediction Error Rate (Accuracy) Comparison with same URL

Sr. No.	URL	Criteria	Error Rate
1	htt:/w33.redifmail.com	Existing(Old Method)-Detection phishing Error-rate	**50.0 %**
2	htt:/w33.redifmail.com	New Criteria(More Feature set)	**14.29%**

B. Clickjacking Attack

Clickjacking attack is a malicious technique to hijack the clicks where victim unintentionally clicks in a web application through attractive offers for winning products. The attacker hides the frame or fake button beside the actual webpage. The proposed work has provided the solutions in detailed experimental setups for Clickjacking attacks [17].

In this exploratory setup, we have tried the use of Clickjacking attack and their prevention on different client machines. The experiment was analyzed for different PCs and the success rate of the attacks without and with any defenses mechanisms, was calculated. The log file of each web site was maintained to generate two files in our database. One file is a counter that increments when a user visits the web page and other file is also counter that increment when a user clicks on the *like* button.

In order to, evaluate the result of Clickjacking prevention before and after applying defense mechanism. The users are visited to web sites and clicks on a hidden button inside an iframe. It provides a confirmation box with warn message for "You have clicked a hidden button". If visitor clicks on the "Yes "button then attack successfully prevention and visitor clicks on "Cancel", the behavior of the hidden button will be discarded. Here, The formula for calculation of the reduced attack success rate after applying prevention mechanism as below.

$$Attack\ success\ rate\ after\ cp = \frac{(Ok\ +\ Cancel)}{Total\ no.of\ visited\ user}$$

The CP (clickjacking prevention) for Google chrome creates a big difference between attack success rates, carried out against without and with defense mechanisms. The attack success rate drastically reduces from 27.27% to 9.09% for websites visited by user and the results are shown as below.

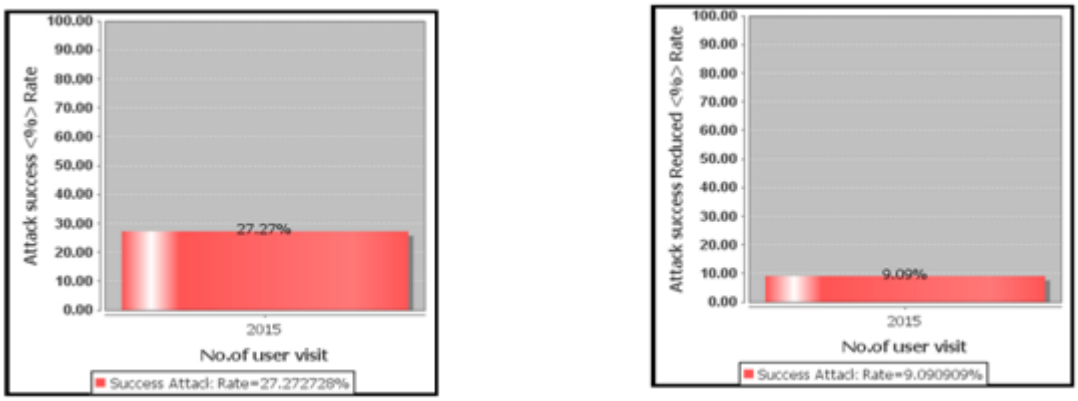

Fig. a: Before Mechanism Fig. b: After Mechanism

Fig. 4: Clickjacking Attack Prevention Result

V. CONCLUSION AND FUTURE SCOPE

There are several website security issues which are common to various types of websites, irrespective of the type of website development language and technology. Analyzing the threats and how that can affect the data and the site itself is an important aspect of web development. Though several techniques and extensions exists in virtually all the technologies, no work presents an interface where both secured and insecure version of the site can be checked. Attackers continue to increase the sophistication of their exploit techniques.

To be secure in the cyber world these days is challenge. In this research work exploration, the demonstration of web based attacks like cookie stealing, phishing and clickjacking and their successful detection and with server side approaches implementation for their prevention. So that identifying the risk on time and mitigating it would lead to a safer browser environment. It also improves fast access of data in any sites due to use of legitimate code in web based programming.

In order to defend against those attacks, we have proposed a web based solution in the form of CP (Clickjack Prevention) that ensures defense against clicking on the embedded sensitive user interface. The CP has effective prevention rate increase up to 50% to 60% for newly proposed Clickjacking attack. Similar, phishing prevention rate

increase 30% than older methods. So our project improves the runtime performance of browser by securing the contents at client side. It may become a more effective, dynamic and interactive type of applications in market.

As a future, the evaluation of web based attacks detection and prevention will dynamically overhead approaches for hosted application at server side using centralized protection mechanism. It may produce effective results for social media sites like Facebook, LinkedIn, Google+ and Twitter. And also it may be adapted for more precisely analyzing JavaScript vulnerability, dynamically in smart phones and other OS for all web browsers.

REFERENCES

[1] Saurabh Jain, Deepak Singh Tomar, Divya Rishi Sahu, "Detection of JavaScript Vulnerability at Client Agen", *International Journal of Scientific & Technology Research,* Vol. 1, August *2012.*

[2] Social Media :http://whatis.techtarget.com/definition/social-media

[3] BrowsersWork: *http://www.html5rocks.com/en/tutorials/internals/howbrowserswork/*

[4] Ram B. Basnets,Andrew H. Sung, Quingzhong Liu, " Learning to Detect Phishing URLs",IJRET, Volume 03,Jun-2014.

[5] TheYearinPhishing,"RSA",January 2013:*http://www.slideshare.net/emcacademics/rsa-fraud-report-january-2013.*

[6] Lin-Shung Huang, Alex Moshchunk,HelenJ.Wang, StaurtSchechter,Collin Jackson "Clickjacking: Attacks and Defenses",2013.

[7] ZALEWSKI, M.Arbitrary page mashups (UI redressing) *https://code.google.com/p/browsersec/wiki/Part2*

[8] Linfeng Li, Marko Helenius, EleniBerki, "A Usability Test of Whitelist and Blacklist-based Anti Phishing Applications", ACM 978-1-4503-1637-8/12/10, pp. 195-202, October, 2012.

[9] L.Breiman and A. Cutler, "Random forests-classification description", Department of Statistics Homepage, *http://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm*

[10] Sebastian Lekies, Mario Heiderich, Dennis Appelt, Thorsten Holz, Martin Johns "On the fragility and limitations of current Browser-provided Clickjacking protection schemes", August 06th 2012.

[11] L.-S. Huang and C. Jackson. Clickjacking attacks unresolved. *http://mayscript.com/blog/david/clickjacking-attacks-unresolved, 2011.*

[12] Marcus Niemietz, "UI Redressing: Attacks and Countermeasures Revisited", RUB 2011.

[13] PravinSoni, ShamalFirake, B. B. Meshram, "A Phishing Analysis of Web Based Systems", ACM 978-1-4503-0464-1/11/02, pp. 527-530, Feb, 2011.

[14] Daehyun Kim and Hyoungshick Kim "We are still vulnerable to clickjacking attacks: about 99% of Korean websites are dangerous", October-2013.

[15] Uhley,Peleus,"Clickjacking Threats", March 2012 : *http://www.w3.org/Security/wiki/Clickjacking_Threats*

[16] Testing for clickjacking: *https//www.owasp.org/index.php/Teseting_for_Clickjacking_%28OTG-CLIENT-009%29*

[17] Ubaid Ur Rehman, Waqas Ahmad Khan, Nazar Abbas Saqib, Muhammad Kaleem, "On Detection and Prevention of Clickjacking Attack for OSNs", IEEE, FIT -2013.

[18] Nilesh Sharma, Nishant Sharma, Vishakha Tiwari, Shweta Chahar,Smriti Maheshwari, "Real-Time Detection of Phishing Tweets", ICCSEA, CSIT -2014.

[19] Sunil Chaudhary,"Recognition of phishing attacks utilizing anomalies in phishing websites", Thesis, Tampere - 2012.

[20] Web Attacks: *http://www.trendmicro.com/vinfo/us/threat-encyclopedia/web-attack/108/think-before-you-click-truth-behind-clickjacking-on-facebook*

[21] HumanCheck*: http://chekkers.atwebpages.com/humancheck.php*

[22] Anupama Aggarwaly, Ashwin Rajadesingan, Ponnurangam Kumaraguru, " PhishAri: Automatic Realtime Phishing Detection on Twitter", eCrime Researchers Summit, 2012.

[23] Clickjacking Defense Cheat Sheet: *https://www.owasp.org/index.php/Clickjacking_Defense_Cheat_Sheet*

[24] Ram B. Basnet, Andrew H. Sung, "Learning to Detect Phishing Webpages", *Journal of Internet Services and Information Security (JISIS)*, volume: 4, number: 3, pp. 21-39.

[25] Mona Ghotaish Alkhozae, Omar Abdullah Batarfi, " Phishing Websites Detection based on Phishing Characteristics in the Webpage Source Code", *IJICT Journal*, Volume 1 No. 6, October 2011.

