ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 3, March 2016

Phish Guard: Detecting Phishing Sites Using Fundamental Visual Similarities

Pritam Dhadve[1], Akshay Matale[2] , Dattatraya Bhaval[3], Vijay Gangad[4], Prof. Pooja Dhule[5]

Marathwada Mitra Mandal College of Engineering, Pune, Maharashtra, India [1,2,3,4]

Assistant Professor, Dept. of Computer Engineering, Marathwada Mitra Mandal College of Engineering, Pune, Maharashtra India[5]

ABSTRACT:In this is paper we represent the concept of the phishing sites detection technique. Phishing is the attack in which the user is not able to know that his/her accessing the legitimate sites or fake sites. for that purpose we are going to provide solution. In this concept we are going to implement the two algorithms Obfuscation Algorithm, CSS Algorithm We are present an algorithm to quantify suspious rating of the webpage based on similarity of visual appearance between the web pages for that we implements the algorithm name CSS algorithm. We present an algorithm to new phishing detection algorithm known as ObURL algorithm .The algorithm detects the DNS Test, IP Address Test, URL Encode Test, Shorten URL Test , White List Test , Black List Test , Pattern Matching Test. It provides the obfuscation phishing attack and it provides multilayer security over the internet .Our intension shows the correctness and accuracy of our goal with relatively low performance overhead.

KEYWORDS: Anti-phishing, iframe, network security, phishing URL Obfuscation.

I. INTRODUCTION

Phishing is the attack of social engineering in which mimicking of the electronic communication to lure user to fill their confidential information. In this user is going to fill their confidential data to the phishing websites.[1] The attacker is going to use that private information for thiefting purpose. In simple word the phishing is mean that sending the email to victim provides the lured information. In this paper we are going to do some types of operation on incoming emails in the inbox and also doing the similarity checks on fundamental visual features. Our assumption is that more than 7000 phishing pages are evaluated. To detect the phishing sites antiphishing is going toused. We are going to provide the solution for securing the most important confidential information from the attacker. Phish Guard is efficiently checks the phishing websites. Phish Guard achieved best result in detecting the phishing websites. It also used for real time phishing detection by the various steps system going to work.

II. LITERATURE SURVEY

[1] Bait Alarm

In this how the CSS algorithm is going to work. In this algorithm included hat page content and layout specification and output the similarity score based on visual features. For each webpages extract page layout and represent it in the normalized of two pages and deciding their similarity

[2] Obfuscation URL algorithm

In this ObURL work into two parts 1) content checking and 2)performing the all operation on webpages or websites .Input as email is send to user than algorithm is going to check the mail content of the email if mail content are match than alert the user give the messages as do not enter confidential information data.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 3, March 2016

The iframe operation is going to start in email content. Algorithms work on multiple iframes and provide multilayer security. After completion of iframe operation than perform all 6 test on URL.
The ObURL algorithm is heuristics algorithms it causes false positive for phishing sites and false negative for non-phishing sites

[3]/ [4] URL Based Heuristics/ PhishStorm
In this various operation on URL features checking the Primary Domain, sub domain, Path domain and the ranking of the sites such as page rank, Alexa rank , Alexa reputation to detecting phishing websites.
Phisher is going to make URL of the phishing websites which look like similar to legitimate sites to fool the users.
Phisher is going to lure their victim by clicking on the wrong URL which is pointing to the phishing websites

[5]Anti-phishing Based on Automated Individual White-List
The black list approach is effective as it maintain all the global phishing sites. We present an approach of white list that contains automated individual white list. It contains all the familiar login user interfaces. If the user are login to the login user interface that is not in the white list it alert the user. The white list also protect against pharming attack because it alert the user when legitimate IP is changed maliciously. As IP address of popular web sites are basically stable. So black list contain all the phishing sites and white list contain all the trusted sites. So this two list are used to protect against the phishing attack by alerting the user.

[6] Identification of Phishing Web Pages
All the phishing web pages have minimum one text field to collect user's confidential information. So the web page contain at least one text field the next test is performed to extract other features. If we page containing text filed for asking personal data, like password or credit cart number of user then we scan HTML for <input> tags that accept text accomplished as "credit card" and "password". We put the websites in the black list that has moved from the one domain to another domain. If age of the domain is greater than 6 months then web site is legitimate one otherwise it is phishing site.

We also check the sub domain and multi sub domain of the web sites. For example http://www.mmcoe.edu.in. A domain name is always included top level domain that is ".in". In this example ".edu.in" is a second domain and "mmcoe" is the actual domain. So we can ignore typing "www.". We are checking that if dots in a domain part are less than three then it is legitimate site. Dots in a domain part are equal to three then it is suspicious otherwise it is phishing site. We are also check for the long URL. If URL length is less than 54 it is legitimate. If URL length greater than 54 and less than 75 then site is suspicious otherwise it is phishing site.The domain record of URL is also checked for security. If no DNS record is found then it is phishing site otherwise it is legitimate.

III. OBFUSCATION DETECTION IMPLEMENTATION

In this paper we introduce an algorithm called Ob-URL detection algorithm. This algorithm provides security against the obfuscated URL. This algorithm is designed to check the hyperlinks, destination URL in hyperlinks, input form, I-Frame in email, I-Frame within I-Frame.[2] We are also going to perfume the tests like DNS test, IP address test, shorten URL test, URL encode test, White list and Black list test and URL pattern matching test on the collected data. Both known and unknown URL obfuscation phishing attacks are detected by this algorithm.

3.1 DNS Test
First hypertext and anchor text are extracted and matched. If both are different then alert the user that both DNS are different.

3.2 IP Test
If the attacker uses IP address instead of domain name then the algorithm checks the IP address with database. Database contains White list and Black list. If IP is same as in White list then it is safe[1]. If IP is same as in Black list then it will alert the user. If IP is not found in both lists then alert the user as IP not found in White list, be careful.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 3, March 2016

3.3 URL Encoded Test
It checks for shortened URL. If URL is shortened by attacker then alert the user.

3.4 Shorten URL Test
Hiding something from user sometime attackers encodes the URL. If URL is encoded then algorithm will alert the user.[2]

3.5 Whitelist and Blacklist Test
In this all the hyperlinks are checked in White list and Black list. If it is found in White list then user is safe. If hyperlink found in Black list then alert the user[5].

3.6 Pattern Matching
The hyperlink and anchor test are checked here. If both are same but not identical then algorithm alerts the user.

IV. CSS ALGORITHM

4.1 Phishing Pages Detection Using Visual Features

The key point of the solution is visual similarity checker of pages. The algorithms split into two parts page content and layout specification and output of this is similarity score of visual similar layout. Cascaded Style Sheet [1] is the web technology that going to use for only in the visual appearance.A CSS rule included two component selector and one more declaration. Selector is split into several categories like tag selector, id selector, class selector, and other tags.

4.2 Avoiding Phishing Attacks

A whitelist in the context of phishing detection is simply a list of trusted websites. For CSS detection to work properly, the list contains more than just the URL of the trusted website. Each entry in the whitelist database contains six strings: the URL of the trusted site, the domain of the site, the title of the site, the CSS filename, the CSS domain, and the CSS content of the file [1].

a **The URL of the trusted site:**
The URL of the trusted site is used to periodically update the CSS information in the database.This is the URL of the site such as “https:\\signin.ebay.com”.

b **The domain of the site:**
The domain of the trusted site is the domain of the URL such as “signin.ebay.com” and is used to determine whether the current page displayed in the browser is on the whitelist or not [1].

c **The title of the site:**
The title of the trusted site is the page title of the site such as “Welcome to eBay” and can be used during CSS content detection to speed up detection by matching potential phishing site titles with titles in the whitelist Database [1].

d **The CSS filename:**
The CSS filename is the filename of the CSS file such as “paypal.css” and can also be used during CSS content detection to speed up detection by matching potential phishing site CSS filenames with filenames in the whitelist database.

e **The CSS domain:**
The CSS domain is the domain of the location of the CSS file such as “secureinclude.ebaystatic.com”.[1] Often the domain is the same as the site domain, but in other cases such as eBay, the CSS file is hosted on a different domain. Storing the CSS domain is essential because if a match is found on a website not in the whitelist, then it is most likely a phishing site linking to the actual CSS file location of the legitimate site.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 3, March 2016

f The CSS content of the file:

CSS content is the actual text contained in the CSS file that contains all of the style information. The CSS content is used to compare with the CSS content of a possible phishing site in order to determine if there is a match with a legitimate site.

Page layout normalization:

To extract features from CSS, we are going convert CSS and page DOM into our presentation.

Step 1:

Rules set extraction
In this step browser captures the CSS structure of the page when user opens the webpage. General representation of CSS is a series of rules as follows

Selector1 {$Property_1$-1:Value1-1; $Property_1$-2: $Value_1$-2; ...};
Selector2 {$Property_2$-1:$Value_2$-1; $Property_2$-2: $Value_2$-2; ...};

Step 2:

In this step CSS rules are converted into the comparison –units

(Comparison-Unit) Given a Web page's CSS rules set, CSS() = {. . . , [$Selector_i${. . . ; [$Property_j$:$Value_k$; . . .], . . .}], . . .}, the corresponding Comparison- Units set of the web page is represented as CompUnit () = {. . . , [$Property_j$: [. . . ;$Value^k_j$:[. . . , $Selector^{j,k}_i$; . . .], . . .], . . . , }

V. FIGURES AND EQUATIONS

5.1 Figures:

Fig.(a) shows the system architecture in which working flow of the system is shown. There are two phases in which both Ob-URL and CSS algorithm executes. In first phase Ob-URL algorithm executes in which all six test as shown in Fig.(a) area performed. This algorithm executes while email fetching. In second phase CSS algorithm executes where all the CSS tags like class tag, ID tag, other tag and HTML tag are checked.

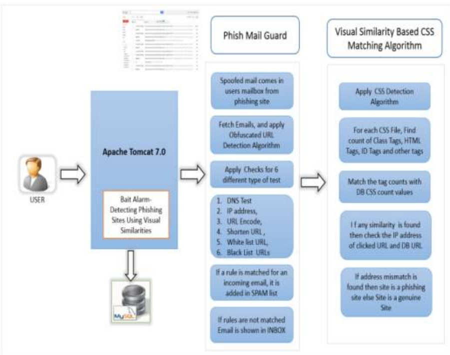

Fig.(a): System Architecture.

ISSN(Online) : 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 3, March 2016

CSS architecture shown in Fig.(b) with example of ICICI bank. This shows how CSS algorithm is applied and how CSS contents are converted into comparison units. This algorithm executes after OB-URL algorithm that is after fetching the emails.

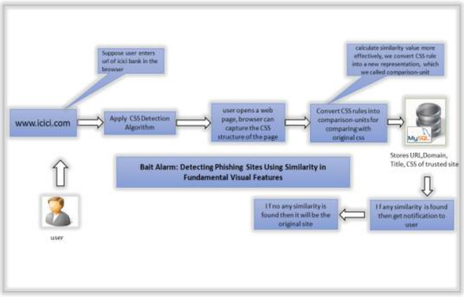

Fig.(b): CSS architecture with example.

Fig.(c) shows how the CSS content are checked. First CSS content are extracted and represented in particular format. CSS checking is done in three levels. First level is block level similarity second level is layout level similarity and last is style similarity.

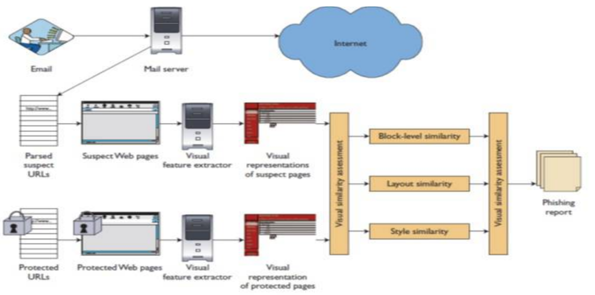

Fig.(c): CSS checking steps

5.2 Equations:

The first equation is complexity score of web pages for fundamental visual layout metrics.

$$S_A = \sum_{n=1}^{N_A} \sum_{m=1}^{M_n} k_t^{n,m} * w_t + k_c^{n,m} * w_c + k_i^{n,m} * w_i + k_o^{n,m} * w_o$$

(a)

Where $N_A$is the number of web page A's properties; $M_n$is the number of the n-th property's optional values {$Value^m{}_n$}; $k^{m,n}{}_t$, $k^{n,m}{}_c$, $k^{n,m}{}_i$and $k^{n,m}{}_o$represent the number of the Tag, Class, ID, Others selectors with the value $Value^m{}_n$respectively and w_t, w_c, w_i, wo are corresponding weight values.

The second equation is Match score of web pages A and webpage B

$$S_{match}^{A,B} = \sum_{n=1}^{N_A} \sum_{m=1}^{M_n} e_t^{n,m} * w_t + e_c^{n,m} * w_c + e_i^{n,m} * w_i + k_o^{n,m} * e_o$$

(b)

where $e^{n,m}{}_t$, $e^{n,m}{}_c$, $e^{n,m}{}_i$and $e^{n,m}{}_o$represent the number of equal selectors with the value $Value^{mn}$belong to the *Tag, Class, ID, Others* categories respectively.

The third equation is of the similarity checker of the webpages A and webpage B.

$$Sim(A,B) = \frac{match\ score\ (A,B)}{min\{score\ (A), score\ (B)\}} = \frac{S_{match}^{A,B}}{min\{S_A, S_B\}}$$

(c)

Comparison of the webpage A and webpage B similarity between webpage A and webpage B Based on this analysis of the CSS algorithm is going to use this type of equation for calculating similarity of webpage A and webpage Similarity score is near about zero. If the similarity score is found near about zero or less than zero then the website is whitelist websites or vice a versa.

VI. CONCLUSION

Phishing is an activity which lures the victim to access that site and enter the confidential information which may use by attacker. Thus we are going to give better solution that effectively prevents websites from phishing attacks using ObURL algorithm which performs DNS test, IP test, Shorten URL test, URL encode test, White list and Black list test and pattern matching test. Our system increases efficiency by using CSS checking which we called CSS algorithm. In the future work, we will work on improving Phish guard's resilience to avoiding attacks by user.

REFERENCES

[1] Jian Mao, Pei Li, Kun Li, Tao Wei, and Zhenkai Liang, "Bait Alarm Detecting Phishing Sites Using Similarity in Fundamental Visual Features", 2013.

[2]Samuel Marchal, Jerome Franc¸ois, Radu State and Thomas Engel,"Anti-Phishing Technique to Detect URL Obfuscation", 2014.

[3] Samuel Marchal, Jérôme François, Radu State, and Thomas Engel, "Phish Storm", 2014.

[4] Luong Anh Tuan Nguyen, Ba Lam To, HuuKhuong Nguyen and Minh Hoang Nguyen,"A Novel Approach for Phishing Detection Using URL based Heuristic", 2014.

[5] Ye Cao, Weili Han, Yueran Le,"Anti-phishing based on Automated Individual White-List", 2011.

[6] Purnima Singh, Manoj D. Patil, "Identification of Phishing Web Pages", 2014.

[7]Nikita Spirin, Jiawei Han, "Survey on Web Spam Detection", 2013.