Fjrs.futa.edu.ng
FUTA Journal of Research in Sciences
ISSN: 2315 – 8239 (Print); E-ISSN: 2489 - 0413

FUTA Journal of Research in Sciences, Vol. 13 (2), October, 2017: 304-321

A WEB ENABLED ANTI-PHISHING SOLUTION USING ENHANCED HEURISTIC BASED TECHNIQUE

[1]O.B Okunoye [1], [2*]N.A Azeez and [3]F.A Ilurimi
[1,2,3]Department of Computer Sciences, University of Lagos, Nigeria.
[1]nazeez@unilag.edu.ng, [2]bokunoye@unilag.edu.ng [3]ellydareal@yahoo.com

ABSTRACT

Phishing is a form of social engineering or website forgery whereby attackers mimic a trusted website or public organization or sending e-mails in an automated manner in order to steal sensitive information or credentials of online users. This is done in a way the user does not realize he is in a phishing environment and in turn reveals his sensitive information such as credit card information, employment details, online shopping account passwords and bank information. Phishers are still having their ways to succeed in their various nefarious activities and attacks. Different anti-phishing schemes however have emerged but phishers still find their ways around by breaking through various existing techniques. Against this backdrop, this project aims at developing a web enabled anti-phishing technique using enhanced heuristic approach. This technique immediately updates the blacklist if a suspicious website is confirmed as a phishing site otherwise considered legitimate and in turn update the whitelist. This novel anti-phishing eradicates the delay in updating blacklist and whitelist. Users will be able to use this web application at will to test if a site is legitimate or not. This technique was implemented using PHP programming language and Database. A total number of Two Thousand Five Hundred and Nineteen URLs were tested (2519) which is represented as "K" while Two Thousand Five Hundred and Ten (2510) were correctly classified and this is denoted as "k. The results after the implementation show that there was no false negative (a phishing URL that is wrongly classified as legitimate) and one false positive (a legitimate URL wrongly classified as phishing). The rate of false positive and false negative is very low when compared with other techniques. Based on the outcome of this work, it is strongly recommended to any company to avoid comprise and to have a reliable & dependable transaction within an organization.

Keywords*:* Phishing, White list, false positive, false negative, Heuristic, blacklist

INTRODUCTION

Internet grants us access to myriads of information and online services as it effectively help service providers to minimize the cost of offering their services. Internet is prone to attack from internet fraudsters who commit electronic online information and identity theft (Ayofe *et al.,* 2010). Phishing is a form of social engineering or website forgery whereby attackers mimic a trusted website or public organization or sending of e-mails in an automated manner in order to steal sensitive information or credentials of online users (Azeez *et al.,* 2015). This is done in a way the user does not realize he is in a phishing environment and in turn reveals his sensitive information such as credit card information, employment details, online shopping account passwords and bank information, etc. Social engineering can take the form of spamming, phishing or scamming that deceive computer users into revealing their confidential information and take actions that are detrimental to these supplied vital information which can be stolen (Khonji, *et al.,* 2013) .

Several attempts have been made to protect online users from this fraudulent financial and informational crime. This has brought about the establishment of Anti-Phishing Working Group (APWG) in 2013. APWG is an international organization that collects phishing information from company contributions, APWG feeds, Anti-Phishing alliance of China (APAC), private sources, etc. and businesses (which could be regional international treaty organizations, government agencies, communication companies (Rao and Ali, 2015), law enforcement agencies, etc.) affected by phishing attacks from different sources. APWG collates and gives the statistics of malicious domain and phishing attacks on a quarterly and yearly basis (Bhandari *et al.,* 2013).

Anti-phishing Working Group collected and analyzed the phishing attack that took place in the first half of 2014 (January 1-June 30) in order to understand trends and significance of phishing attack by quantifying the scope of the global phishing problem. Anti-phishing Alliance of China (APAC), China Internet Network Information center (CCNIC), private sources and several phishing feeds provided supplement to the data collected by APWG (Azeez and Ademolu, 2016). This has made it possible for APWG to have a large and comprehensive repository of email fraud and phishing activity (APWG Global Phishing survey report 1H2014). APWG reported that on monthly basis, there is 50% increase in phishing attack of which 5% of the phishing emails lure users to visit phishing site. (Hiba , 2014) systematically reviewed previous and current research done on internet phishing in different fields and their approaches to solve phishing attacks and provided information that can be used to fill the gaps that exist in terms of security in these approaches. According to Mishra,, (2014), from the evaluation report done on different anti-phishing techniques, it was concluded that no algorithm can be considered best in phishing detection as the performance of various methods used focus on particular target application case since attackers can change their tactics with little or no cost.

LITERATURE REVIEW

Phishing Methods

Various sources used in perpetuating phishing have been identified in the Literature review. These sources are described below:

Website based phishing- Phishers target users to reveal their sensitive information by duplicating a trusted website, users enter their sensitive details (password, credit card information, social security number or personal information) thinking it is an authentic website. According to Kirda and Kruegel, (2006), the phishers use corporate identity, real logos from original website, obfuscated URLs and host names and model in such a way that inexperienced users cannot detect it to be illegitimate website. They also make use of Javascript code, hidden images and frames to control the way page is displayed by the browser of the victim (Azeez and Iliyas, 2016).

Malware based Phishing-In this case, a legitimate or trusted website may be compromised and malicious software (such as viruses, worms, Trojans & spyware) inserted via a video, audio file or link, once a user clicks on the link, the malicious software is installed which runs malicious codes on user's computer in order to steal users sensitive or confidential data such as

passwords, software activation keys, etc. without the user knowing or aid phishing techniques (Kiran *et al.*, 2013).

E-mail based Phishing -This is the earliest form of phishing attack. In this case, several numbers of e-mails claiming to originate from a reliable source are sent to millions of users in which a large number of users could fall for it. The phisher alters part of the e-mail header or sender address so that users can fall into the trap (James and Philip, 2012).

Types of Phishing

There are different types of phishing, some of which are discussed below

Sphear Phishing-In this kind of phishing, the phisher target selected class of people that have something in common, companies or organizations to steal intelligent information, military information or business secret (Azeez *et al.*, 2011). The phisher monitors the frequent visit of the user to a particular legitimate site and then compromise that site in order to make users vulnerable to their attacks (Lenny, 2016). Unlike general phishing which is aimed at committing financial theft by casting millions of emails randomly, Spear phishing target a particular group of people. Whaling which is a type of Spear phishing target the "Keyman" (CEO or Directors) of an organization. The Canadian Government, Oak Ridge National Laboratory, HBGary Federal, and Australian Prime Minister's office were victims of sphear Phishing attack in late 2010 and early 2011 respectively (Hiba , 2014).

Clone Phishing- The attacker clone a legitimate email by using the information from the original email such as recipient address and content. The attacker replaces the link with a malicious one and resend to the recipient in which the receiver will think it is another version of the previous email from the original sender (Yue *et al.*, 2007).

Phone Phishing- this type of phishing is done in form of messages coming from a financial institution requesting users to call a phone number regarding issues with their bank details or to visit a particular link in order to update some vital information about their account detail (account number, credit card number & pin) to keep their account active (Azeez and Babatope, 2016). Attackers use infrastructures like Internet Relay Chat (IRC) or instant messaging systems such as ICQ to lure users to visit fake websites (Kirda and Kruegel, 2006).

Spoofing-This is a means of fraudulently hacking network to have unauthorized access to steal identity of a trusted website and makes some modifications so that messages sent by the phisher will appear as coming from a legitimate source. This is done with the aim of hijacking sensitive information and this form of attack is perpetuated using email, website or through calls. Spoofing can be in the form of ARP poisoning, DNS spoofing, IP address spoofing and web spoofing (Khonji *et al.*, 2013).

Anti-Phishing Approach- Anti-Phishing is a process that aims at protecting by warning users directly and effectively from entering their sensitive information into an unsafe website (Azeez and Venter, 2013). Web browsers such as Mozilla Firefox, Google Chrome, Internet Explorer (IE), and Opera now integrate anti-phishing application. Some of the anti-phishing approaches are discussed below:

Heuristic Approach-It uses website contents, URL signatures or HTML to detect phishing websites and identify phish. It extracts features of a phishing after studying the features of the phishing URL and website content and based on the extracted features, it design a means of detecting the phishing websites (Rao and Ali, 2015). Heuristic can produce true positive and true negative rates and has the ability to detect the moment an attack is launched. The disadvantage with this approach is the fact that it can incorrectly label a legitimate site as a phish by producing false positive (Jyothi *et al.*, 2013). It is also possible for a phisher to bypass heuristics and actually achieve the aim of stealing confidential information or financial fraud (Gglosser, 2008).

Visual Similarity based Approach-This approach is not fast as it has to make comparison between authentic domain visual content and a suspicious website. A particular threshold is set, if the similarities between the compared visual

content is above the set threshold then it is regarded as a phish (Nureni and Irwin, 2010).

Blacklist Approach-This approach check URLs against respective list of known phish. The blacklist may contain the domain or IP addresses used by known phishers but fails to detect if the URLs fall out of list. The blacklist can be hosted at a central server or stored locally at the client server (Azeez and Lasis, 2016). It is easy for phishers to evade because blacklist solely rely on the exact match with entries already blacklisted. Blacklist has certain limitations: With this approach, attacker can modify top level domain (TLD) to URLs. Unless updated, blacklist cannot detect new threat from phisher since phishing sites are short-lived (Mao, 2013). Blacklist does not consume large resources on the user machine

Anti-Phishing techniques such as Heuristic, Visual similarity and blacklist and whitelist approach are becoming less effective due to the availability of sophisticated phishing toolkits for fresh phisher. This has increased the phishing techniques of sophisticated phishers (Aaron and Rasmussen, 2013). To combat the growing phishing attack, there is need for more effective anti-phishing technique that incorporates all the anti-phishing approach. According to Mishra (2014), an effective anti-phishing technique must possess four basic requirements (Mishra, 2014):

I. It should be able to differentiate a spoofing from phishing
II. Must be able to match various attacks from respective sources
III. Give convincing evidence/example of copying
IV. Should be able to identify the target organization of the phisher.

Whitelist Approach-In this case, the suspicious URLs is checked against existing list of known legitimate sites. If the URL is present then it is considered legitimate site otherwise it is considered phishing site. Based on the drawback experienced in blacklist which is not having the complete list of global phishing sites, this led to the development of Automated Individual White List (AIWL) (Enoch *et al.*, 2013). AIWL keeps track of all familiar Login user interfaces (LUIs) of website in a whitelist. If the white list does not contain the LUI and the user is about to enter some personal information, it warns the user of the potential danger of possibly entering into a phishing zone. It also alert user if the legitimate IP is maliciously tampered with or adjusted and it can detect pharming and phishing attacks (Yue *et al.*, 2007).

ANTI-PHISHING TECHNIQUES

Phishnet Anti-phishing Technique uses two major components namely: URL prediction and Approximate URL matching component.

In URL prediction component, it uses different heuristics to create a new URL from a known phishing URL, analyses and then proceed to test whether the created URL is actually malicious by using five heuristics. The five heuristics used are as follows:

- It replaces the Top Level Domain (TLD)
- It checks if several phishing URLs have directory structure similarities with few variations
- If the URLs have directory structure similarities but the domain name are different and they point to the same IP address, then it is classified a match
- It divides several variations of URL with diverse query strings
- It checks for the equivalent brand name (Prakash *et al.*, 2010).

The second component which is approximate URL matching uses algorithm to break a simple URL into several parts, which is matched with existing blacklist (Prakash *et al.*, 2010). PhishNet is language-independent that uses heuristics, blacklist and image based to detect phishing attacks. It cannot detect zero day phishing attacks and does not use white list approach to detect phishing site (Azeez and Venter, 2012).

PhishShield Anti-phishing Technique is novel heuristic solution that detects phishing attacks using five level of detection. It takes URL as input and produces the status of the site as unknown, phishing or legitimate as output. Phishshield first compares the URL of a domain against the list (white list) of legitimate websites, if it fails that test then the URL is considered a phishing else regarded as a legitimate site, the HTML webpage

is saved as a DOM element(Document Object Model) and then passes to the next level of detection (Miller, 1998). A DOM is a language-independent interface that defines the logical structure of documents and the way those documents are manipulated and accessed. A login page can be found through parsing the HTML of site for input type known as "password" (Azeez *et al.,* 2011). If this password type field is absent, then it is classified as phishing and stops the execution process but if not, the execution process continues after which it goes to the next level of detection (known as zero link in the body portion of HTML (Kiran *et al.,* 2013).

This level searches for at least a link which must be present in the body of a legitimate webpage. If the HTML body has zero number of links, it is considered as phishing. The third level of detection calculates the value of links of a footer link present in the website (Lakshmi and Vijaya, 2012). If it points to null then it is tagged as phishing but if not it is passed to the next level of detection to check copyright and title content and compared with white list for any match after extraction to verify if it is a legitimate site (Liu *et al.,* 2006). The last level of detection look out for and compares the frequency of hyperlinks that points to its own domain to the one pointing to foreign domain. If the frequency of hyperlinks pointing to its own domain is higher than the hyperlinks pointing to foreign domain, then it is classified as legitimate else as phishing. PhisShield is language-independent that uses heuristics that can detect image based phishing attacks; zero day phishing attacks and uses white list to compare URLs but does not use visual similarities and blacklist to detect phishing attacks (Rao & Ali, 2015).

The aim of this work is to extend the work already done on PhishShield by adding Whois lookup, SSL and updating blacklist & whitelist. The algorithm used for developing PhishShield could be found in (Rao and Ali, 2015) and its architecture is depicted in Figure 1.

From the algorithm (Source: (Rao and Ali, 2015), PhishShield takes URL as input. It checks if the URL is in the whitelist and if it is not it uses some heuristics to detect phishing, these are zero links in body of html, footer links with null value, copyright content, title content and website identity (consider the frequency of each domain in links of each webpage and get the one with maximum frequency domain in order to identify the targeted site) and outputs the status of URL as legitimate or phishing website. JSoup is used to extract html contents and manipulate data (contents) of the webpage like CSS footer, etc. and to prevent XSS attacks (Liu *et al.,* 2006).

APPROACH TO IMPLEMENTATION

This research work enhances the work of Phish shield (Rao and Ali, 2015) by incorporating blacklist, SSL and "whois" lookup into the application. Anti-Phishing is an application that aims at protecting by warning users directly and effectively from entering their sensitive information into an unsafe website. Phish Detect is a web based application that uses certain features to classify a site either as legitimate or phish site. Based on these criteria, if anyone of the features is missing, then phish Detect classifies the site as a phish else it is classified as legitimate site and update the blacklist and white list respectively (Liu *et al.,* 2006).

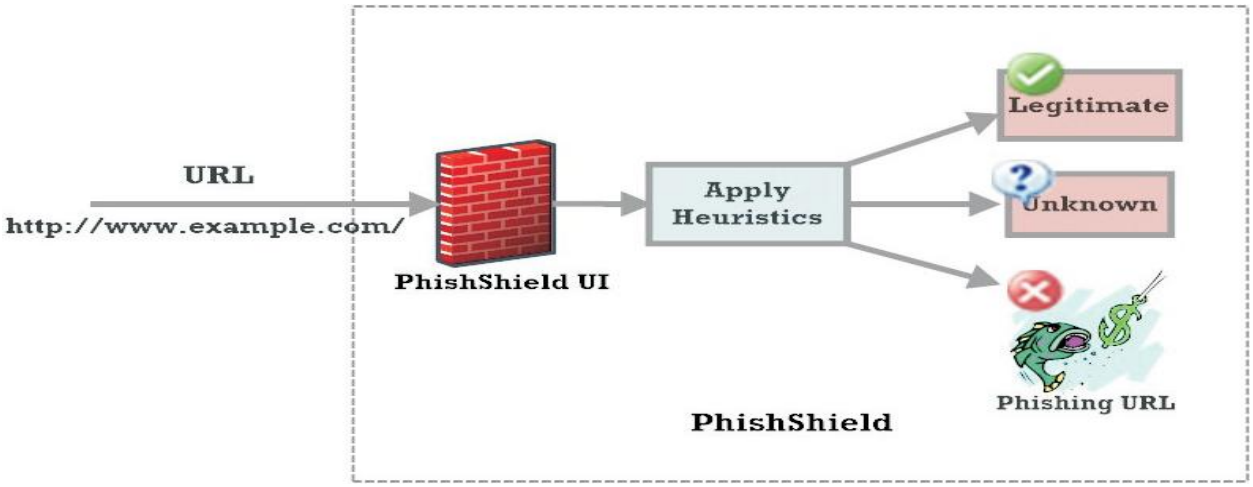

Figure 1. Architecture of PhishShield Anti-Phishing Technique (Rao and Ali, 2015).

URL Signature

A user is expected to submit a suspicious URL to the application, this updates the tested URL database, the application checks if the submitted URL is in the whitelist repository, if not present, Phish Detect checks if the URL is a valid string and uses PHP regular expression to check the URL if it has (http, https,: , //, [a-z0-9-]) but if the string is not valid, the URL is added to the UNKNOWN database . It further checks for any suspicious character in the URL base on RFC 1738 specification (" <> # % { } | \ ^ ~ [ ]) which when present, the application classifies the URL as phishing and in turn checks the phishing repository if URL is present, if not the phishing database is updated (Azeez, 2012).

HTTP Ok Code 200

Phishdetect checks the URL if it passes the Http ok code 200 (which means the URL is valid) and then moves to the next stage of the test. If the http ok code 200 returns as false then phishing repository (blacklist) is consulted to check if the URL is already present but if not the URL is updated on the phishing (blacklist) database.

Apply Heuristics

Scan for Link in the Page Body

The application scans through the downloaded page to look out for the number of links in the page body. If there are no links, Phish Detect flag "The site has phishing attribute" otherwise it passes the test for that stage and goes to next stage. In the case of phishing site, image is used to replace page content without a link in the page body of the HTML and it redirect the user to a foreign anchor whereas such is not the case with legitimate website i.e. frequency of the hyperlinks pointing to its own domain high. A legitimate site must have headlines and hyperlink text either to submit contact form, buy product, sign up, forgot password or others with unique website contents that are up-to-date (Shahriar and Zulkernine, 2011).

Cascading Style Sheet (CSS)

Cascading Style Sheet (CSS) allows the control of layout and look of a page. It tags or properties are easy to use and affect the use and feel or style of that page in terms of design, screen sizes, variations in display for various devices and layout. Phish Detect scans through the downloaded webpage source to know if pages contain display and rendering information (e.g., fonts, colors, spacing). This is an indication that the webpage is compatible with multiple browsers. Although this does not further prove that a URL is phishing or not, but it checks if cross-site requests are allowed. If not found then there is a very good chance webpage is phishing when cross checked with other variable (Ye Cao *et al.*, 2008).

Footer Links Pointing to NULL (#)

Phish Detect scans through the downloaded webpage source to know if it is pointing to NULL (#) value. This is an indication that the link is redirecting to its own page. If there is such, the application considers it as phishing site else it is classified legitimate.

The footer link pointing to NULL (#) can be in any of these forms:

<tr id=”footer” value=”#”></tr>

<tr id=”footer” value=”#content”></tr>

<tr id=”footer” value=”#skip”></tr>

Title and Copyright Content

Title and copyright content in a legitimate site often contains the details of the domain of the website. The contents of the copyright extracted are broken into tokens (smaller fractions). The fraction is compared with the whitelist and there occur any match, Phish Detect flags “it has attributes of phish”, which means it is classified a phishing site.

Secured Socket Layer Certificate (SSL)

SSL certificate is used to create a secured connection between a web server and a user browser for protected or encrypted transmission of sensitive information (e.g. login credentials, credit card number, social security number, etc.) without third party tapping into or intercepting the information. Phish Detect look out for these feature as every legitimate website should have SSL certificate (Ye Cao *et al.,* 2008). In the case of phish website, phishers can use a fake SSL in which the pointed URL starts from shttp: // or https:// or that a legitimate website should possess. SSL certificate is issued by SSL certificate authority (CA) and this requires three keys to set up a secured connection between a web browser and a user browser. The three keys are private key, public key and session key (Liu et. al., 2006).

Use of Third Party

Whitelist

This is a list of legitimate websites. If URL is submitted for query, the whitelist repository is first consulted to check if the suspicious URL submitted is on the list. If the URL is classified legitimate after Phish detect has scan through, the whitelist is updated.

Blacklist

This is a DNS- based anti-phishing approach that contains list of suspected or phishing website. If the submitted URL is scanned by the proposed application and it is detected as a phishing website, then URL is updated on the blacklist or phishing database.

Use of Whois Lookup

Whois contains the list of registered site with the information of the registrar and registrant. It provides the details of the website registration date, last update and expiration date. Phishing site do not last for so long and so the phishers do not tend to register their sites (Ye Cao *et al.,* 2008). For the purpose of this work, it is expected that all legitimate website be on whois database. Phish Detect look this up on the Whois server and if the website is not registered, then it is classified as phishing site.

ENHANCED HEURISTIC APPROACH ALGORITHM

Algorithm 1. Enhanced Heuristic Approach Algorithm

START

Input*: an URL*

Output: label (legitimate =0, phishing =1, unknown =2.

1. ***ValidateWebsite(String URL)*** *//validate URL using whitelist*

1.1 Domain=Extract_Domain (URL);
1.2 For each Host name (Host) in Whitelist
1.3 Status=Compare (Host, Domain)
1.4 If (status) return 0;
1.5 Else goto step 2

2. int PhishShield (String URL)

*2.1 Document Doc= Jsoup.Connect (String URL) /*Parse the html of website using JSoup and store the content in a document object Doc*/*
2.2 If (Parsing== Successful) //JSoup Connection is successful
2.2.1 If (Doc has input type==password) //validate login
2.2.1.1. int label=0;
2.2.1.2. ***ImageBasedPhishing (Document Doc)****//Check for number of links (n1) in the body of html*
2.2.1.2.1. n1 = doc.body().select ("a");
2.2.1.2.2 If (n1! = 0) goto step 2.2.1.3
2.2.1.2.3 elselabel = 1; //indicating image phishing website.
2.2.1.2.4 goto step 2.2.1.5;
2.2.1.3. ***NullFooterLinks (Document Doc)*** *//Check the number of footer links equalling to null*
2.2.1.3.1. Elements f1= doc.select("div[id=bottom[footer] a");
2.2.1.3.2 for each link in f1
2.2.1.3.3 f2=checkforNullLinks(Elements f1)// we compared each link with '#' value
2.2.1.3.4. if(f2==0) goto step 2.2.1.4
2.2.1.3.5 else label= 1 // indicating phishing sites having footer links to null
2.2.1.3.6. goto step 2.2.1.5;
2.2.1.4. CopyrightTitle (Document Doc)*//Extract the copyright and title section from html*
2.2.1.4.1. Tokenize the copyright or title section content

2.2.1.4.2 Compare each token with whitelist

2.2.1.4.2.1. if comparison successful

2.2.1.4.2.2 label = 1

2.2.1.4.2.3. elsegoto step 2.2.1.5

2.2.1.5 . WebsiteIdentity (Document Doc) *//calculate the frequency of each domain in links of the webpage each and*

2.2.1.5.1.Webidentity=CalculateDomainwithMaximumFrequency (Doc); / we counted frequency of found maximum frequency domain*/*

2.2.1.5.2. If (domain of input URL! = web identity)

2.2.1.5.3. then label = 1 *//i.e. input URL is targeting website identity.*

2.2.1.6. return label

2.2.2. *Else Stop Executing PhishShield Application// case of absence of password field*

2.3 *Else return 2; // case of parsing failure*

Check if URL passes HTTP ok 200 code

Else flag URL is phishing

SSL in the web page body

Check for SSL in the page

If SSL is present, go to the next step

Else check if URL is on blacklist *//i.e. URL is in phishing repository*

Whois lookup

If URL is present on whois database *//i.e. URL is registered*

Flag as URL is legitimate

Update the Legitimate database *//i.e. URL is not already on the legitimate repository*

Else flag URL is Phishing

Update the Phishing database *//i.e. URL is not yet blacklisted*

STOP

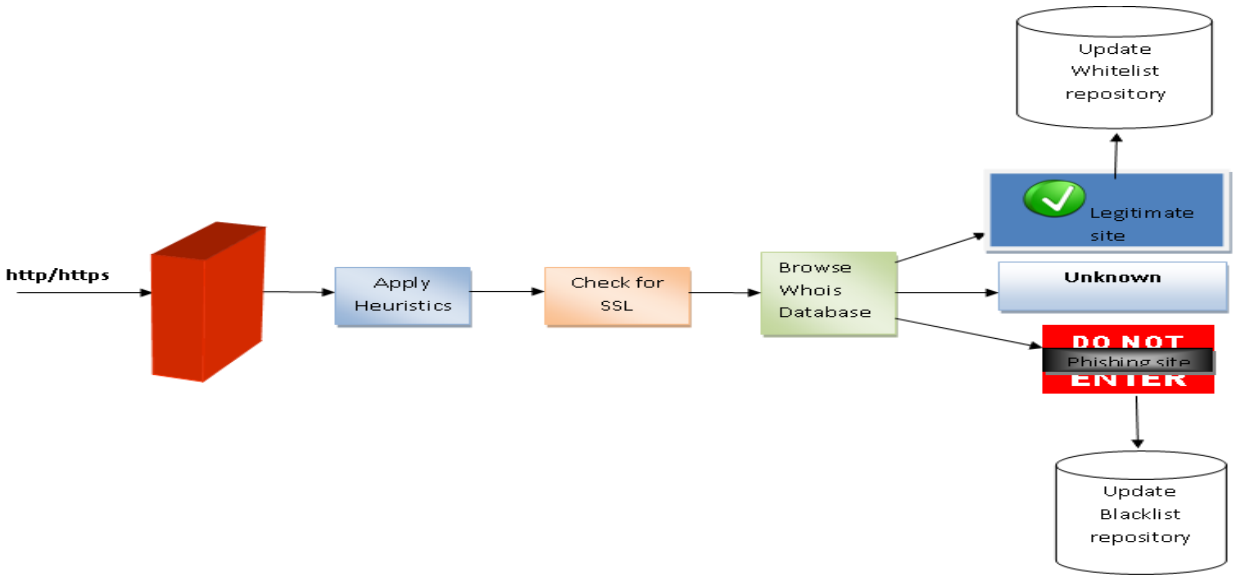

Figure 2. Architecture of PhishDetect Anti-phishing Application

RESULT AND EVALUATION OF RESULT

Legitimate Site: The URL is entered into the query box of PhishDetect, the application looks up the URL. If the URL passes the Http ok code 200, the application further checks the content of the page body (title, CSS, links, image, CSS), if these features are present it checks for the SSL. If the URL does not pass the SSL check test due to the fact that not all legitimate site have SSL especially when it is not a login platform where payments are made or confidential information are transferred, then the application consult the "Whois" database to confirm if the site is registered. The presence of the URL on the "whois" database indicates it is a legitimate site.

Phishing Site: When a URL is entered into PhishDetect, the application run through the submitted URL, if the URL does not have the features of a legitimate site, the application classifies it as a phishing site. The Figure 5 below is the screenshot of a URL that tested positive (i.e. phishing site). The URL only has a title whereas other attributes of a legitimate site are not present and as such it is classified as a phishing URL. The phishing repository is updated.

Graphical Representation of PhishDetect Output

The Figure 3 below shows that when the total number of tested URLs were 9, the application returned 6 URLs as legitimate which are true negative, 2 URL as phishing that is true positive and 1 unknown which the application could not classified as either legitimate or phishing.

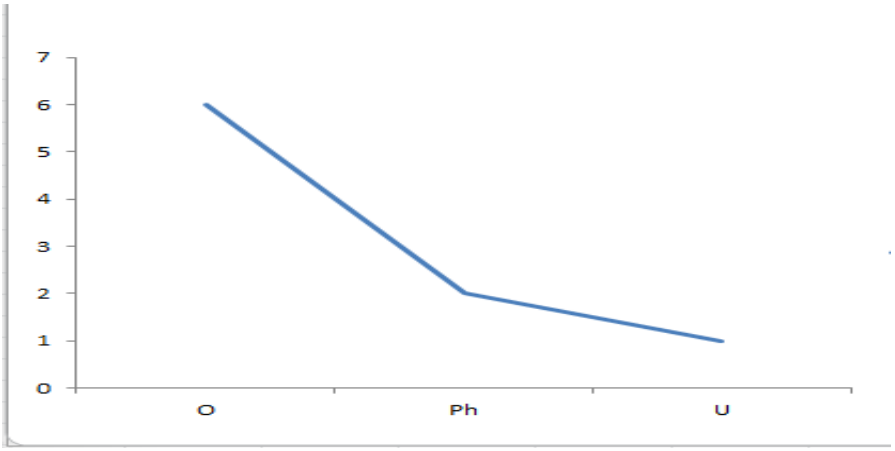

Figure 3. Graphical of Tested URLs at 9 plotted against Legitimate, phishing & unknown URLs

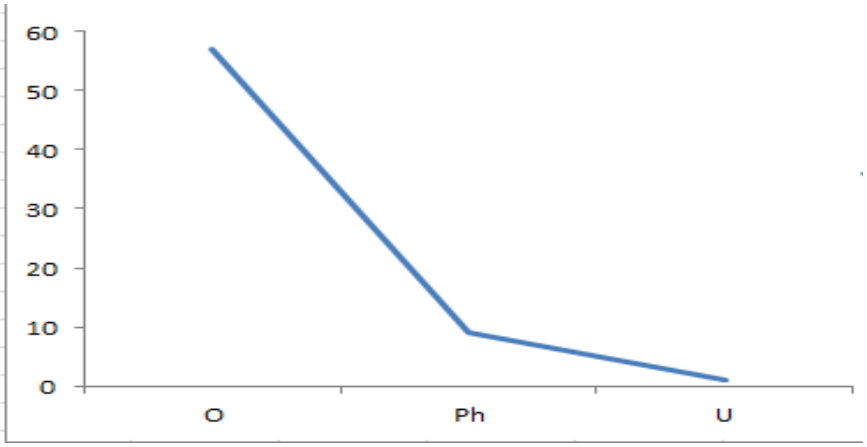

Figure 4. Graphical representation of Tested URLs at 67 plotted against Legitimate, phishing & unknown URLs

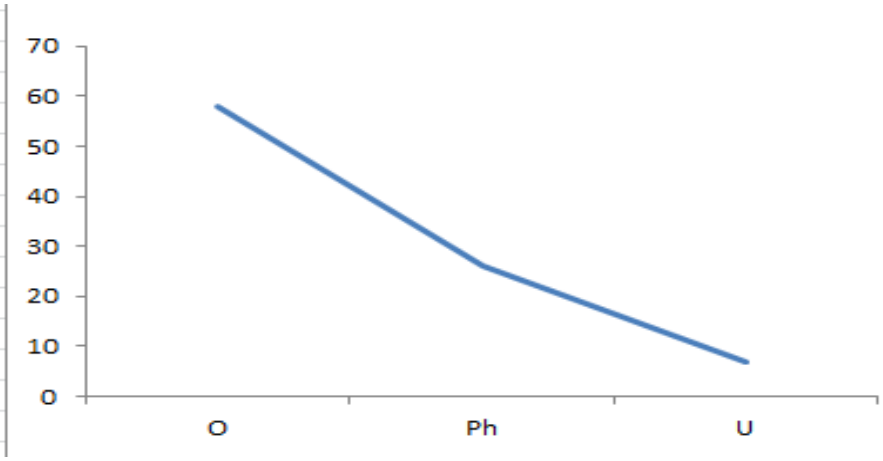

Figure 5. Graphical representation of Tested URLs at 94 plotted against Legitimate, phishing & unknown URLs

When the number of tested URLs was increased the application detected more phishing URLs than legitimate URLs based on the input. The graph of Figure 4 shows that when the total tested URLs was at 67, the application detected 57 URLs correctly as legitimate, 9 URLs correctly classified as phishing and 1 URL as unknown in which the application could not classify it as either phishing or legitimate since it possess the features of both phishing and legitimate URL.

When the number of tested URLs increased to 94, the application correctly classified 58 URLs as legitimate, 26 URLs correctly classified as phishing and 7 URL as unknown. Detailed graphical representation is in Figure 5

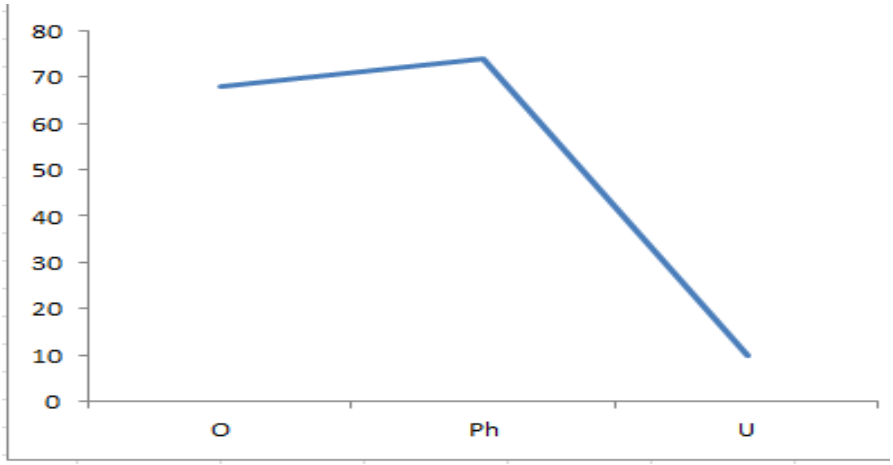

Figure 6. Graphical representation of Tested URLs at 169 plotted against Legitimate, phishing & unknown URLs

When the number of tested URLs increased to 169, the application correctly classified 68 URLs as legitimate, 74 URLs correctly classified as phishing and 10 URL as unknown. This is depicted in Figure 6.

As the total number of tested URLs increases to 273, the application correctly classified 152 URLs as phishing, 101 URLS were classified as legitimate and 11 URLs were classified as unknown. This is shown in Figure 7.

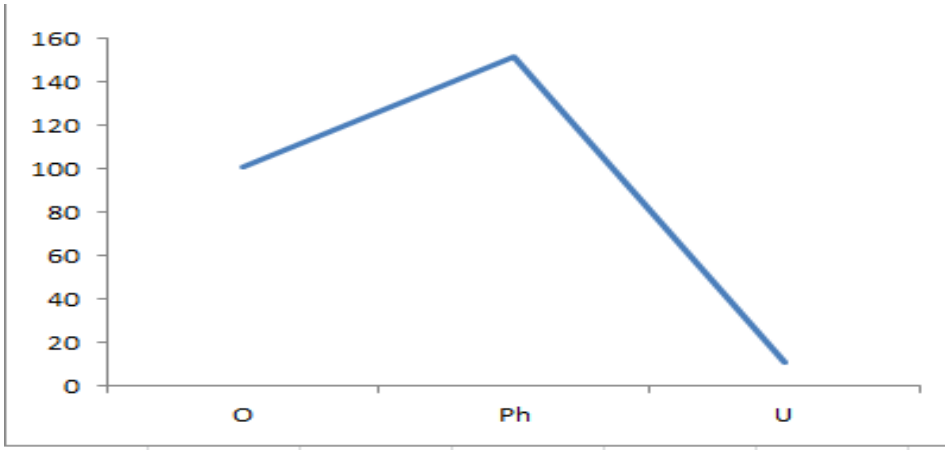

Figure 7. Graphical representation of Tested URLs at 273 plotted against Legitimate, phishing & unknown URLs

ANALYSIS OF RESULTS

The following analyses are done to test and know the effectiveness of the application's output result. K is the total number of submitted URLs while k is denoted as the number of correctly classified URLs.

Table 1. Experimental Results

	Legitimate URLs	Phishing URLs	Total
K	102	151	253
k	101	151	252

Table 2. Experimental Results

	Legitimate URLs	Phishing URLs	Total
K	200	268	468
k	198	268	466

Table 3. Experimental Results

	Legitimate URLs	Phishing URLs	Total
K	250	300	550
k	248	300	548

Table 4. Experimental Results

	Legitimate URLs	Phishing URLs	Total
K	282	310	592
k	280	310	590

Table 5. Experimental Results

	Legitimate URLs	Phishing URLs	Total
K	300	356	656
k	298	356	654

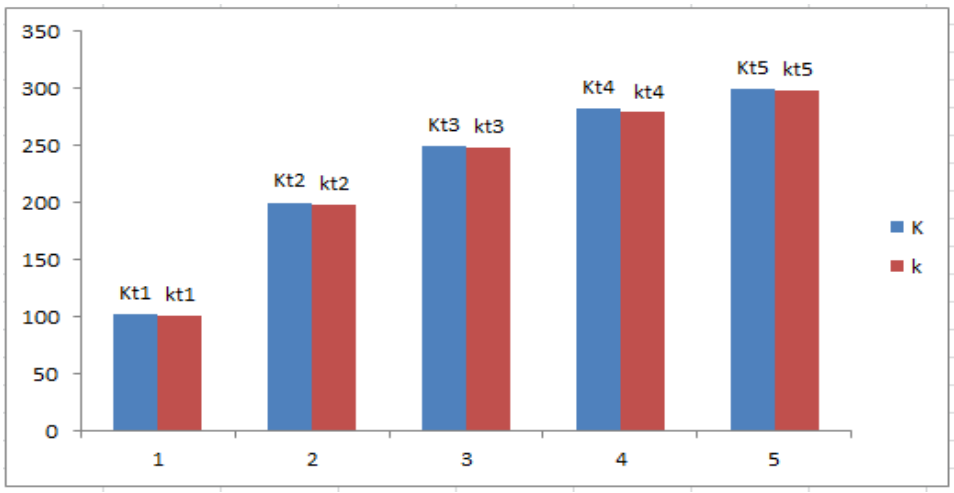

Figure 8. Graph of Total Number of Submitted Legitimate URLS Plotted Against Correctly Classified Legitimate URLS

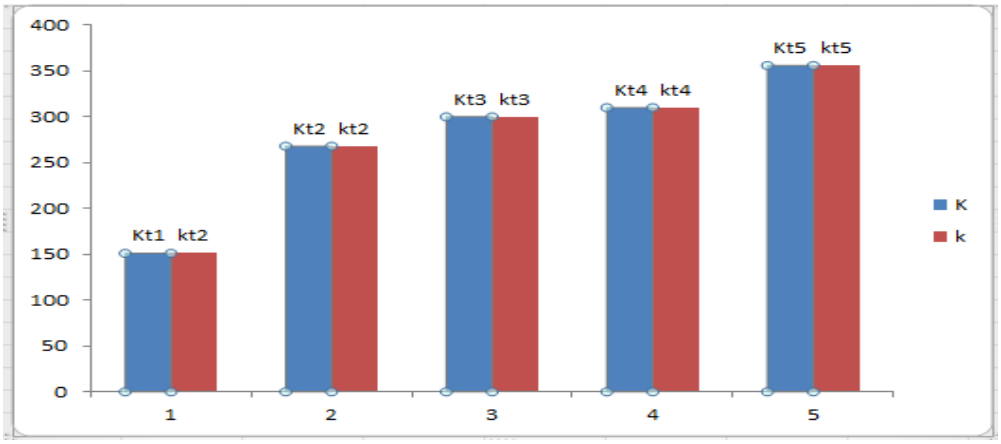

Figure 9. Graph of Total Number of Submitted Phishing URLS Plotted Against Correctly Classified Phishing URLS

Evaluation Metrics

The metrics for the evaluation are:

True Positive Rate (TPR): This measures the rate of phishing sites (P_h) that are correctly classified as phishing sites (P_h). It is denoted as:

$$TPR = \frac{P_h \rightarrow P_h}{P_h + F_{neg}} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

Where F_{neg} = the number of phishing site wrongly classified as legitimate site

$$TPR = \frac{151}{151+0}$$

TPR =1*100 =100%

True Negative Rate (TNR): This is the measure of the rate of legitimate sites (O) that is correctly classified as Legitimate sites (O). It is denoted as:

$$TNR = \frac{O \rightarrow O}{O + F_{pos}} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

Where F_{pos}=the number of legitimate site wrongly classified as phishing site

$\text{TPR} = \frac{101}{101+1}$, $\text{TPR} = \frac{101}{102}$, TNR = 0.99*100= 99%

False Positive Rate (FPR): This measure the rate of legitimate sites (O) falsely classified as phishing sites (P_h). It is denoted as:

$$\text{FPR} = \frac{O \rightarrow P_h}{F_{pos} + T_{pos}} \quad \ldots (3)$$

Where T_{pos}=the number of phishing site correctly classified as phishing site

FPR= $\frac{1}{1+151}$ = $\text{FPR} = \frac{1}{152}$,

FPR= 0.0066*100 = 0.7%

False Negative Rate (FNR): This measure the rate of phishing sites (P_h) wrongly classified as legitimate site (O). It is denoted as:

$$\text{FNR} = \frac{O \rightarrow P_h}{F_{neg} + T_{neg}} \quad \ldots (4)$$

Where T_{neg} = the number of legitimate sites correctly classified as legitimate sites

$\text{FNR} = \frac{0}{0+101}$ = FNR = 0*100 = 0%

Accuracy (Acc): This is the measure of overall rate of classified sites in relation to the sum of the actual or correctly classified legitimate sites and phishing sites. It is denoted as:

$$\text{Acc} = \frac{(P_h \rightarrow O)}{P_h + T_{pos} + O + T_{neg}} \quad \ldots (5)$$

Acc= $\frac{151+101}{151+0+101+1}$, $\text{Acc} = \frac{252}{253}$, Accuracy (Acc) = 0.996*100 = 99.6%

Table 6. Phishing Data Source

Source	Sites	Link
Phishtank's open database	110	http://www.phishtank.com/phish_archive.php
Other sources	30	
Public Block Lists of Malicious IPs and URLs	40	http://www.selectrealsecurity.com/public-block-lists
DNS-BH – Malware Domain Blocklist		56 http://www.malwaredomains.com/?page_id=66
LENNY ZELTSER - Blocklists of Suspected Malicious IPs and URLs	37	https://zeltser.com/malicious-ip-blocklists

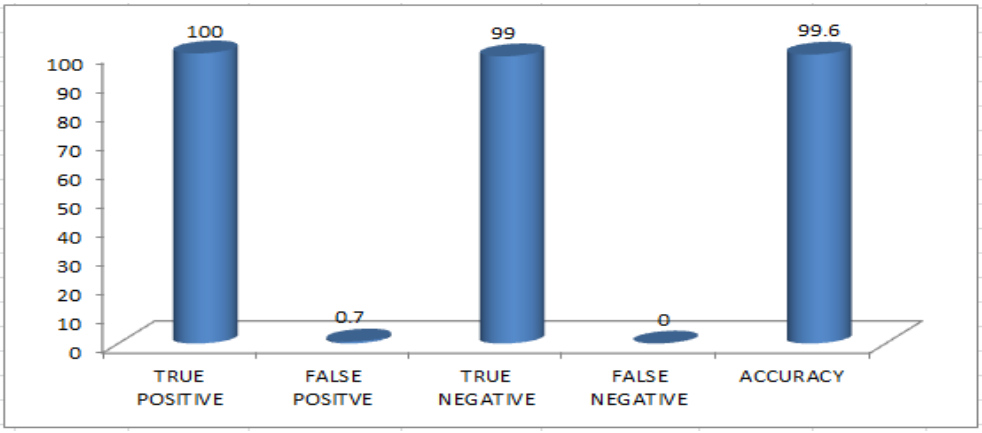

Figure 10. Performance Result of PhishDetect Application

Graphical Representation of PhishDetect Application Performance

The effectiveness of the application is shown below. From the graph, which is depicted in Figure 10, the application is able to detect correctly phishing and legitimate URLs with very minimal false positive and false negative rate.

CONCLUSION

This research work has presented a novel method to detect phishing sites. This application is able to detect phishing site in 5 steps. The application first check the URL if the URL passes HTTP ok code 200, which means the URL is complete and valid. The second step is to check for suspicious characters in the URL that are peculiar to phishing URLs and if this characters are present, then the URL is classified phishing and moved into the phishing repository.

In the third step, the application checks the content of the webpage and look out for the presence of Title, CSS, Links, and Images. If all this features are present it further checks for SSL which may not necessary be present in some legitimate sites. Lastly the application consults "whois lookup" database to verify if the website is a registered website in order to correctly classify the URL as either legitimate or phishing. URLs that possess features of peculiar to phishing and legitimate site are updated in the Unknown database as they cannot be classified by the application. This approach is web-based that uses the content of the webpage, SSL and whois database for classification. It requires little or no previous knowledge of the website. It is user friendly. Future work can be done on the application as it does not use visual similarities approach as this approach is time consuming in terms of the response time and the targeted site by the phisher.

RECOMMENDATION

It is believed that this approach will help companies improve the online business platform and save online users from the heartache caused by phishers from stealing their personal and confidential information to defraud them. With this approach, users can verify any suspicious URL in and avoid falling victim of cyber-crimes.

ACKNOWLEDGEMENT

The authors wish to acknowledge the efforts of anonymous referees for their valuable comments and helpful suggestions in shaping this paper into a publishable condition.

REFERENCES

Aaron, G. & Rasmussen, R., (2013). *APWG: Unifying the global response to crime*, USA:

Global Phishing Survey 2H2013, Trends and Domain Name Use.

Ayofe, A.N, Adebayo, S.B, Ajetola, A.R, Abdulwahab, A.F (2010) "A framework for computer aided investigation of ATM fraud in Nigeria" International Journal of Soft Computing, Vol. 5, Issue 3 pp. 78-82

Azeez, N.A, Olayinka, A.F, Fasina, E.P, Venter, I.M. (2015) "Evaluation of a flexible column-based access control security model for medical-based information" Journal of Computer Science and Its Application. Vol. 22, Issue 1, Pages 14-25

Azeez, N. A., and Ademolu, O. (2016). CyberProtector: Identifying Compromised URLs in Electronic Mails with Bayesian Classification. 2016 International Conference Computational Science and Computational Intelligence (CSCI) (pp. 959-965). Las Vegas, NV, USA: IEEE.

Azeez, N. A., and Babatope, A. B. (2016). AANtID: an alternative approach to network intrusion detection. The Journal of Computer Science and its Applications. An International Journal of the Nigeria Computer Society, 129-143.

Azeez, N. A., and Iliyas, H. D. (2016). Implementation of a 4-tier cloud-based architecture for collaborative health care delivery. Nigerian Journal of Technological Development, 13 (1), 17-25.

Azeez, N. A., and Venter, I. M. (2013). Towards ensuring scalability, interoperability and efficient access control in a multi-domain grid-based environment. SAIEE Africa Research Journal, 104 (2), 54-68.

Azeez, N. A., Iyamu, T., and Venter, I. M. (2011). Grid security loopholes with proposed countermeasures. In E. Gelenbe, R. Lent, and G. Sakellari (Ed.), 26th International Symposium on Computer and Information Sciences (pp. 411-418). London: Springer.

Azeez, N.A., and Lasisi, A. A. (2016). Empirical and Statistical Evaluation of the Effectiveness of Four Lossless Data Compression Algorithms. Nigerian Journal of Technological Development, Vol. 13, NO. 2, December 2016, 64-73.

Azeez, N. A. (2012). Towards Ensuring Scalability, Interoperability and Efficient Access Control In a Triple-Domain Grid-Based Environment. Cape Town: University of the Western Cape.

Azeez, N.A and Venter, I.M (2012). Towards achieving scalability and interoperability in a triple-domain grid-based environment (3DGBE)- Information Security for South Africa (ISSA), 2012, pp 1-10.

Bhandari, M., Wale, S. & Gayatri, M., (2013). Anti-Phishing Approach using Probabilistic (t,) VC Scheme. *International Journal of Advanced Research in Computer Science and Software Engineering*, 3(10), pp. 821-826.

Gglosser, (2008). DNS-BH – Malware Domain Blocklist, [online] Available at: http://www.malwaredomains.com/?page_id=66 [Accessed 20April 2016].

Enoch, Y. S., Adebayo, K. J. & Olumuyiwa, A. E., (2013). Mitigating Cyber Identity Fraud using Advanced Multi Anti-Phishing Technique. *International Journal Of Advanced Computer Science and Applications*, 4(3), pp. 156-164.

Hiba Z., (2014). Current State of Anti-Phishing Approaches and Revelation Competencies. *Journal of technological and Applied Information technology*, 70(3), pp. 507-515.

James, D. & Philip, M., (2012). *A novel anti phishing framework based on visual cryptography*.. IEEE., In Power, Signals, Controls and Computation (EPSCICON), 2012 International Conference on (pp. 1-5)..

Jyothi, Y. Y., Srinivas, D. & Govindaraju, K., (2013). The Secured Antiphishing approach using image based validation. *International Journal of Research in Computer and communication Technology*, 2(9), pp. 796-801.

Khonji, M., Youssef, I. & Jones, A., (2013). Phishing detection: a literature survey.. *Communications Surveys & Tutorials, IEEE*, 15(4), pp. 2091-2121.

Kiran, V. K., Gowtham, R. & Archanaa, R., (2013). An RDF Based Anti-Phishing Framework. *International Journal of Software and Web Sciences (IJSWS)*, 6(1), pp. 1-10.

Kirda, E. & Kruegel, C., (2006). Protecting users against phishing attacks.. *The Computer Journal*, 49(5), pp. 554-561.

Lakshmi, V. & Vijaya, M., (2012). *Efficient prediction of phishing websites using supervised learning algorithms.* s.l., Procedia Engineering, 30, pp.798-805., pp. 798-805.

Lenny Zeltser - Blocklists of Suspected Malicious IPs and URLs, (2016). [Online] Available at: https://zeltser.com/malicious-ip-blocklists [Accessed 30 April 2016].

Liu, W. Deng, X., Huang, G. & Fu, A. (2006). An antiphishing strategy based on visual similarity assessment. *IEEE Internet computing*, 10(2), p. 58.

Mao, P. L. (2013). Detecting Phishing Sites using Similarity in Fundamental Visual Features. *In 5th International Conference on Intelligent Networking and Collaborative Systems*, Issue INCoS 2013, IEEE, p. 790–795.

Miller, E., (1998). *An Introduction to the Resource Description Framework,* Dublin, Ohio: D-Lib Magazine.

Mishra, V. (2014). A Survey of Various Anti-Phishing Techniques. *Universe of Emerging Technologies And Science*, 1(1).

Nureni, A. A., and Irwin, B. (2010). Cyber security: Challenges and the way forward. Computer Science & Telecommunications, 29, 56-69.

Prakash, P., Kumar, M., Kompella, R. & Gupta, M., (2010). *Phishnet: predictive blacklisting to detect phishing attacks.* IEEE, In INFOCOM, 2010 Proceedings IEEE (pp. 1-5), pp. Pages 346-350.

Rao, R. & Ali, S., (2015). *PhishShield: A Desktop Application to Detect Phishing Webpages through Heuristic Approach.*. s.l., Procedia Computer Science, 54, pp.147-156., pp. 147-156.

Shahriar, H. & Zulkernine, M., (2011). Trustworthiness testing of phishing websites: A behavior model-based approach.. *Future Generation Computer Systems*, 28(8), pp. 1258-1271.

Ye Cao, Weili Han & Le, Y. (2008). *Anti-Phishing Based on Automated Individual White-List.* USA, Resaerch Gate.

Yue, Z., Jason, H. & Cranor, L., (2007). CANTINA: A Content-Based Approach to Detecting Phishing Web Sites. *Banff, Alberta, Canada.,* Volume ACM 978-1-59593-654-7/07/0005.