Phish-Shelter:
A Novel Anti-Phishing Browser Using Fused Machine Learning

Rizwan Ur Rahman, Maulana Azad National Institute of Technology, Bhopal, India
Lokesh Yadav, Maulana Azad National Institute of Technology, Bhopal, India
https://orcid.org/0000-0001-7206-2794
Deepak Singh Tomar, Maulana Azad National Institute of Technology, Bhopal, India

ABSTRACT

A phishing attack is a deceitful attempt to steal confidential data such as credit card information and account passwords. In this paper, Phish-Shelter, a novel anti-phishing browser, is developed. It analyzes the URL and the content of phishing page. Phish-Shelter is based on combined supervised machine learning model. Phish-Shelter browser uses two novel feature sets, which are used to determine the web page identity. The proposed feature sets include eight features to evaluate the obfuscation-based rule and eight features to identify search engine. Further, the authors have taken 11 features that are used to discover content and blacklist-based rule. Phish-Shelter exploited matching identity features, which determines the degree of similarity of a URL with the blacklisted URLs. Proposed features are independent from third-party services such as web browser history or search engine results. The experimental results indicate that there is a significant improvement in detection accuracy using proposed features over traditional features.

KEYWORDS

Anti-Phishing Browser, Network Security Attack, Phishing, Phishing Detection, Sensitivity Analysis, URL Matching Identity

INTRODUCTION

With the passage of time, the usage of internet is growing both for individual users and for the organizations. It has become an integral part of our day-to-day social and financial activities. Many of the organizations such as Amazon, Paytm and Myntra offer online trading and online sales of services and goods.

With the increase of front end applications to access the information, internet banking creates the necessity to use reliable methods. In the current scenario, the financial crimes are replaced from direct to indirect attacks. For example, a bank's client could be targeted with a specific trick instead of a robbery (Philippsohn, 2001).

With the increase of the usage of internet, the internet community is much more vulnerable to security attacks. The network security attacks are primarily physical, syntactic, and semantic attacks (Ashton, 2017).

The physical attacks are committed against physical piece of equipment for instance, hard drives, routers, or other electronic devices.

DOI: 10.4018/JITR.2022010104

The Syntactic attacksmay be grouped under the term malware or malicious software. These attacksmay consist of worms, viruses, and Trojan horses. Syntactic attacks, where networks and operating logic are targeted for example web bot attack Trojan and Denial of Service (Rahman et al., 2012).

And finally, Semantic attack is a type of attack, which directly targets the end users instead of physical device and software application. Instead of taking advantage of system vulnerabilities, semantic attacks make use of the way humans interact with computers or interpret messages. Semantic attacks target user-computer interface with the intention of deceive a user into performing an action that will breach a system's information security (Heartfield et al., 2017).

Recently, the most common semantic attack that has been seen is phishing. Phishing is an identity theft which makes use of both social engineering and fake web-site creating methods issued to deceive user to disclose his/her secret and valuable details. Phishing attacks take advantage of user's inability to differentiate between legitimate company websites and fake websites.

In phishing, a semantic attacker uses an email message which appears to be from a legitimate business, such as a bank . The messages look similar to the official one, and can contain html links which leads to a website that resembles legitimate business website. The attackers offer some service via this html link.

Anti-Phishing Work Group (APWG) that is a non-profit organization functioning to provide anti-phishing education to improve the public understanding of security. China Internet Network Information Center (CNNIC), Anti-phishing Alliance of China (APAC) and private sources across the world (APWG, 2012).

APWG produces and releases reports in quarterly, half yearly and yearly describing the statistics of malware and malicious domains and phishing attacks in different constituencies of the world.

To Detect Phishing Attacks, till date many different methods have been proposed. According to APWG, defense mechanisms used for phishing attacks are divided into three methods:

- Content Based Technique
- Heuristic Based Technique
- Blacklist Based Technique

Content Based technique inspects the similarity between the original and spoofed web pages to identify web spoofing. One of the main content based techniques is CANTINA (Zhang et al., 2007) which is successful in the identification of phishing website but it disable the keyword extraction.

Heuristic based approach uses HTML or URL signature to identify the spoofed web-pages. Number of researches conducted based on this approach. One of the main heuristics approach solutions used is SpoofGuard. It is anti-decision maker traffic in checking URL characteristics of phishing web-pages. It also extracts URL Characteristics of phishing browser plugins.

Blacklist Based Approach has been widely used over a long time and it has been adopted as an anti-phishing solutions. This approach contains an updated blacklist for the known phishing Websites. All the entries that are denied access are contained in the phishing blacklist (sheng et al., 2009).

In summary, contributions of this paper are as follows:

- Phish-Shelter, a novel anti-phishing browser is developed in this paper.
- The proposed model introduced feature sets including eight features to evaluate the obfuscation-based rule, and eight features to identify search engine. Further, we have taken eleven features which are used to discover contents, and blacklist based rule.
- Proposed model is evaluated with real websites samples and websites from www.phishtank.com.

RELATED WORK

Currently numerous anti-phishing solutions are available, but most of them are not intelligent enough to make a precise decision, as a result the false-positive decisions increased heavily. In this section, outlines of some of the anti-phishing methodologies and the features they provide in developing anti-phishing solutions are explored. Table 1 presents comparison of phishing related works with proposed work.

An approach that identifies phishing by recognizing the visual characteristics of a suspicious websites was proposed by (Maoet al.,2016). The visual features such as layout similarity Document Object Model (DOM), block level (text and images) and overall style (cascaded style sheet) are compared with respective features of original website. All features are assigned with certain weights as per the priority used while designing a legitimate website. Based on the threshold value the websites are categorized as phishing website or legitimate one. The disadvantage of this method is that it takes high response time i.e. this method requires a large legitimate image database and then visually compares suspicious website with image database which is too costly (Liuet al.,2006).

The two most widely used techniques for defense against phishing attacks are the blacklist and the heuristic based (Aaron & Manning, 2012; Sadeh et al., 2007). In the blacklist approach, the requested URL is compared with a list of predefined phishing URLs. The drawback of this approach is that it does not deal with all phishing websites since a recently launched fake website takes a considerable amount of time before being added to the list. On the other hand, the heuristic-based approach can identify recently created fake websites in real-time (Aaron & Manning, 2012; Sadeh et al., 2007).

CANTINA method identifies web spoofing by inspecting the similarity between the legitimate and spoofed web pages. The web page content is used to calculate the similarity between the two web pages. This method has significant accuracy and low false alarms in identifying the fake web page. CANTINA conducted one research which belongs to this approach. This research identifies the phishing websites by using Term Frequency/Inverse document Frequency (TF-IDF).

Sumner and Yuan introduced, the prevention of social engineering and phishing attacks, using education and training techniques.

Using TF-IDF technique and text mining to retrieve information reduces the false positive rate. The results that we get from CANTINA research shows that it detects about 97% phishing-sites with approximately 6% false positive.

The PhishGuard tool developed by (Joshi et al., 2008) also detect phishing websites during the login process of a website by providing actual credential after the bogus credential. They have also proposed architecture to determine if the website is authorized or a phished one.

In 2010, an intelligent system has been provided to identify phishing pages in e-banking .This model is based on the combination of fuzzy logic with data mining algorithms to investigate techniques by classifying the phishing types and to characterize the e-banking phishing website factors (Hossain. et al., 2010).

Another technique was proposed by (Liu et al., 2013) that identifies phishing based on the visual features of a suspicious websites. These visual features such as layout similarity (DOM), block level (text and images), and overall style (cascaded style sheet) are compared with respective features of real website.

Hara et al. developed a technique based on image similarity to classify the dubious websites. Author compares the authorized and dubious image using ImagSeek application. This technique automatically updates the white-list by adding the suspicious websites that are characterized as neither authorized nor phishing. The bottle neck of this approach is very high false positive and high false negative rate. At client side image comparison leads to delay in browser's experience (Hara et al., 2009).

Abraham et al. demonstrate the design, implementation, and evaluation of hostile to phishing email classifier that procedures lexical URL analysis (LUA) as a classification feature. The essential thought process behind extending the procedure of lexical URL analysis into the email classification

Table 1. Comparison of phishing related works with proposed work

Work	Method	Zero day protection	Search engines Independent	Language independent
Zhang et al., 2007	CANTINA	Yes	No	No
Joshi et al., 2008	New method	Yes	N/A	Yes
Hossain et al., 2010	Fuzzy logic	No	Yes	Yes
Ramesh et al., 2014	New approach	Yes	No	Yes
Rami et al., 2014	Rule-based	Yes	Yes	Yes
Thabtah et al., 2014	Rule-based	Yes	Yes	Yes
Zhang et al., 2014	New method	Yes	Yes	No
Abraham et al., 2014	String matching based	No	Yes	Yes
Peng et al., 2018	Natural Language Processing	N/A	Yes	No
Rao & Pais, 2018	Machine Learning	N/A	No	Yes
Patil et al., 2019	Hybrid Method	N/A	No	Yes
Nagunwa et al., 2019	Machine Learning	Yes	No	Yes
Proposed work	**Hybrid approach**	**Yes**	**Yes**	**Yes**

territory is that utmost phishing email messages enclose phishing URLs, and thus scrutinizing them can deliver classifiers with additional discriminative features that can improve their classification accuracy (Abraham et al., 2014).

Peng et al., 2018 developed an approach which exploits NLP (natural language processing) method to examine the text and identify improper statements which point outs phishing attacks. Rao & Pais, 2018 implemented classification system using heuristic features which are extracted from three sources i.e., source code, URL, and third-party services to overcome the limitations of existing anti-phishing techniques.

Patil et al., 2019 proposed hybrid method which exploited three well known approaches i.e., blacklist, heuristics and visual similarity. Nagunwa et al., 2019 proposed a framework for classifying zero day phishing websites by proposing new hybrid features. The result analysis of the features was analyzed using eight learning algorithms in which Random Forest algorithm achieved the best.

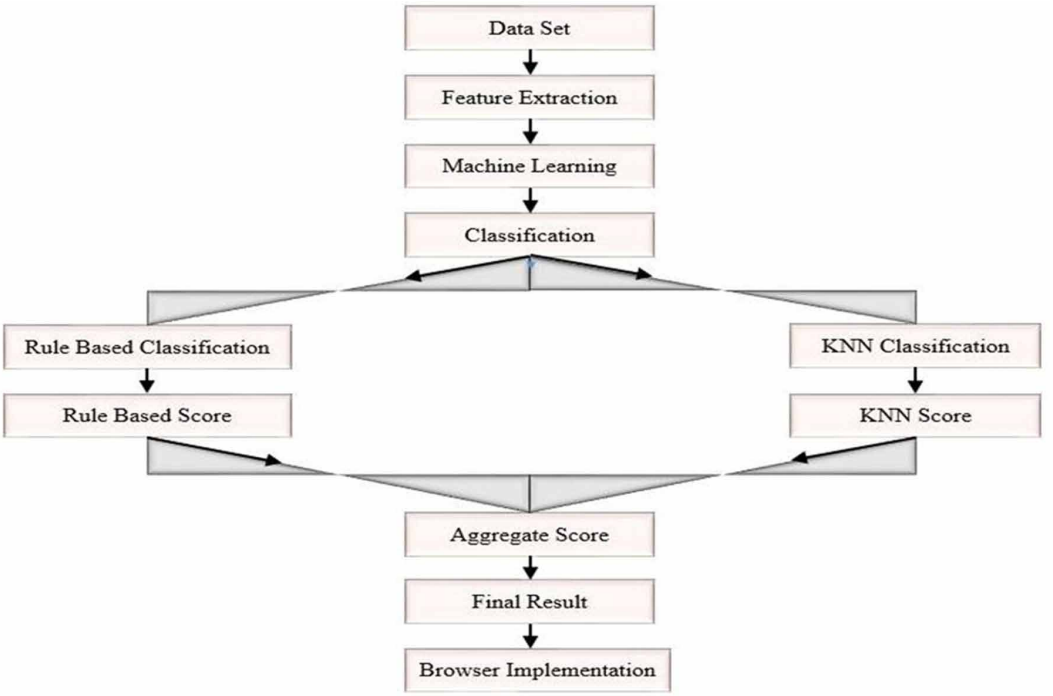
However, in all of the current phishing detection approaches have two common limitations. Firstly, methods based on blacklist are not enough capable to detect zero day phishing attacks. Secondly, existing solutions mostly rely on the contents of HTML pages but in reality the phishing pages are obfuscated.

In this paper, an attempt is made to give solution to the above mentioned problems. In this paper hybrid method is developed for phishing detection, which uses two novel feature set, used to determine the webpage identity. The proposed feature sets include eight features to evaluate the obfuscation-based rule, and eight features to identify search engine. The proposed model for phishing detection uses two machine learning techniques. 1) Rule based method; and 2) K-nearest Neighbor classification algorithm. The two learning techniques balance each other in phishing detection. Rule based learning is more precise to identify known phishing attacks, while the K-nearest Neighbor is more specific to identify unknown phishing attacks.

PROPOSED METHODOLOGY

In the following section, an overview of how proposed system is being used to detect phishing is presented. The proposed system is based on hybrid supervised machine learning approach to improve the performance of detecting phishing attacks and preventing data loss in internet banking web-pages, email, and online social media. The overall system architecture of proposed approach is shown in Figure 1.

Figure 1. Architecture of proposed model



The proposed work motivates on categorizing the relevant features that discriminate phishing websites from legitimate websites and then subjecting them to classification data mining. In order to detect the relevant features, some statistical investigations and analysis were carryout on the phish tank (http://www.phishtank.com) and legitimate dataset.

The system proposes a scheme for phishing page detection based on two phases:

1. Rule Based Feature Set
2. URL Matching Identity Feature Set

Rule Based Feature Set

In this section a novel taxonomy of Rule based Features for phishing website detection has been proposed. According to this taxonomy, the Rule Based features set are classified into the following categories

- Obfuscation based rule
- Search engine and key word based rule
- Abnormality based Rule
- Content and blacklist based Rule

Obfuscation Based Rule

1. Enumerate Prefix or Suffix to the URL distinct by (-) hyphen

Legitimate URLs rarely use dash symbol. Prefixes or suffixes separated by (-) are added to the domain name by the attackers to make users feel that they are dealing with a legitimate webpage for example, http://www.sbi-online.com/.

$$R_1 : URL = \begin{cases} If\left(Url\,Part\,having\ \left(-\right)Symbol\right) \rightarrow PhishingURL \\ else \rightarrow LegitimateURL \end{cases}$$

2. Having the IP Address

When a domain name in the URL contains IP address, such as www.205.53.73.105/fake.html", it indicates that someone is stealing user's personal information. In some cases, to confuse the users, the IP address is transformed into hexadecimal code such as

"http://www.0x82.0xBC.0xCB.0x42/2/xyz.ca/index.html".

$$R_2 : URL = \begin{cases} If\left(The\,Domain\,Part\,having\,the\,IP\,Address\right) \rightarrow PhishingURL \\ else \rightarrow LegitimateURL \end{cases}$$

3. Lengthy URL to Hide the Dubious Part

Attackers use lengthy URL to hide the suspicious part in the address bar.

$$R_3 : URL = \begin{cases} if\ \ (URL\,length < 54) \rightarrow LegitimateURL \\ else\,if\left(URL\,length \geq 54\,and \leq 75\right) \rightarrow SuspiciousURL \\ else \rightarrow PhishingURL \end{cases}$$

To ascertain legitimacy in the study, the lengths of URLs in the dataset are calculated and then produced as an average URL length. It is summarized that if the length of the URL is greater than or equal to 54 characters, then URL is classified as phishing.

4. Position of the Last Occurrence of "//" " in the URL

Double forward slash "//" within the URL path will redirect the user to another website. An example of URL's with "//" is: "http://www.legitimate.com//http://www.phishing.com". The location where the "//" appears is examined. "//" should appear in the sixth position if the URL starts with "HTTP". However, "//" should appear in the seventh position if the URL starts with "HTTPS"

$$R_4 : URL = \begin{cases} If\ \ (Position\,of\,the\,Last\,Occurrence\,of\ \ "//"\ \ in\,the URL > 7) \rightarrow Phishing \\ else \rightarrow Legitimate\,URL \end{cases}$$

5. Having Sub Domain and Multi Sub Domains

Consider the following link: https://www.nita.ac.in/students/. The domain name might include the country-code top-level domains (CCTLD), as ".in" in an example. ".ac" stands for "academic", the combined "ac.in" is known as second-level domain (SLD) and the actual name of the domain is "nita". Rule for extracting this features include removal of (www.) from the URL which is in fact a sub domain in itself followed by removal of (CCTLD) if it exists. Finally, the remaining dots are counted. The URL is classified as "Suspicious" if the number of dots are greater than one, since it has one sub domain. However, it is classified as "Phishing" if the dots are greater than two, since it shall have multiple sub domains. If the URL has no sub domains, then "Legitimate" is assigned to the feature.

$$R_5 : URL = \begin{cases} if\left(Dots\,In\,Domain\,Part = 1\right) \rightarrow Legitimate\,URL \\ else\,if\left(Dots\,In\,Domain\,Part = 2\right) \rightarrow Suspicious\ \ URL \\ else \rightarrow Phishing URL \end{cases}$$

6. By URL Shortening Services TinyURL

URL can be made smaller in length and still can lead to the required webpage on the "World Wide Web" using a method called URL shortening. An "HTTP Redirect" on a domain name, which is short and links to the webpage that has a long URL can be use to achieve this. For example, the URL"http://manit.ac.in/index.php?option=com_content&view=article&id=507&Itemid=238"can be shortened to "https://tinyurl.com/loujxxy".

$$R_6 : URL = \begin{cases} If\left(Tiny URL\right) \rightarrow Phishing URL \\ else \rightarrow Legitimate\,URL \end{cases}$$

7. For URL's having "@" Symbol

Usage of "@" symbol leads the browser to ignore everything preceding it and the real address often follows it.

$$R_7 : URL = \begin{cases} If\left(Url\,Having\ @Symbol\right) \rightarrow PhishingURL \\ else \rightarrow LegitimateURL \end{cases}$$

8. Having Non-Standard Port

It is used to validate if a particular service is up or down on a specific server. It is better to merely open ports that you need to control intrusions. By default, most of the ports will be blocked or only selected ones are open by several firewalls, Proxy and Network Address Translation (NAT servers. Attackers can run almost any service needed if all ports are open.

$$R_8 : URL = \begin{cases} If\left(\text{Port no.is of the } Preffered\,Status\right) \rightarrow PhishingURL \\ else \rightarrow LegitimateURL \end{cases}$$

1. Search engine and key word based rule

a. Based on domain Registration Length

The phishing website lives for a short period of time and trustworthy websites are regularly paid for several years in advance. From the dataset, it was found that the longest domain have been used for one year only.

$$R_9 : URL = \begin{cases} If\left(Domains\,Expires\,on \leq 1 years\right) \rightarrow PhishingURL \\ else \rightarrow LegitimateURL \end{cases}$$

b. Age of Domain

WHOIS database is used to extract this feature. Phishing websites lives for a short period of time. Six months is the minimum age of the legitimate domain.

$$R_{10} : URL = \begin{cases} If\left(Age Of\,Domain \geq 6 months\right) \rightarrow PhishingURL \\ else \rightarrow LegitimateURL \end{cases}$$

c. For DNS Record

The claimed identity of the phishing websites are not recognized by the WHOIS database or no records founded for the hostname. The websites are classified as “Phishing” if the DNS record is empty or not found; else it is “Legitimate”

$$R_{11} : URL = \begin{cases} If\left(no\,DNS\,Record\,For\,The\,Domain\right) \rightarrow PhishingURL \\ else \rightarrow LegitimateURL \end{cases}$$

d. Having Website Traffic

The popularity of the website can be determined by the number of visitors and the number of pages they visited. Phishing websites are not recognized by the Alexa database as they are short lived. It could be found from the dataset that the legitimate websites are ranked among the top 100,000 even in the worst scenarios. Furthermore the domain is classified as "Phishing" if the domain has no traffic or is not recognized by the Alexa database, else as "Suspicious".

$$R_{12} : URL = \begin{cases} if\ \ (WebsiteRank < 100,000\ ) \rightarrow LegitimateURL \\ elseif(WebsiteRank > 100,000) \rightarrow Suspicious \\ else \rightarrow PhishingURL \end{cases}$$

e. For Page-Rank

The importance of a webpage on the internet is measured by Page-Rank and the value ranges from "0" to "1" greater the rank, more the importance. Analysis of our datasets showed that no Page-Rank was found for 95% of the phishing web-pages and the Page-Rank value of the remaining 5% of phishing web-pages may reach up to "0.2".

$$R_{13} : URL = \begin{cases} if\left(PageRank < 0.2\right) \rightarrow PhishingURL \\ else \rightarrow LegitimateURL \end{cases}$$

f. Search Index

It examines the presence of the website in the Google's index. Sites indexed by google will be displayed on the search results. Many phishing web-pages may not be found on the Google index as they are merely accessible for a short period.

g. Server Form Handler (SFH)

Certain SFHs containing an empty sting or "about: blank" are considered doubtful as an action should be taken upon the submitted information. Also, if the domain name in SFHs and the domain name of webpage differ, then the webpage is considered suspicious as the submitted information is rarely handed by external domains.

$$R_{15}: URL = \begin{cases} if\left(SFH\, is\ \text{"about: blank"}\, Or\, Is\, Empty\right) \rightarrow phishing URL \\ else if\left(SFH\ \text{Refers To}\ A\, Different\, Domain\right) \rightarrow suspicious \\ else \rightarrow legitimate \end{cases}$$

h. Submitting Information to Email

Personal information submitted by the user in the web form is directed to the server for processing. The user's information is redirected by the attacker to his personal email. To implement this, server-side script might be used such as "*mail()*" function in PHP. Client-side function such as *mailto*: function may also be used.

$$R_{16}: URL = \begin{cases} If\left(Using\ \text{"mail()" or "mailto:" Function to Submit User Info.}\right) \rightarrow Phishing \\ else \rightarrow Legitimate URL \end{cases}$$

2. Abnormality Based Rule

a. Request URL

The external objects contained within a webpage such as images, videos and sounds are examined by Request URL to check if they are loaded from another domain. The webpage address and most of objects embedded within the legitimate webpage are shared from the same domain.

$$R_{17}: URL = \begin{cases} if(\%of Request URL < 22\%) \rightarrow Legitimate \\ else if\left(\%of Request URL \geq 22\% and\, 61\%\right) \rightarrow Suspicious \\ Otherwise \rightarrow feature = Phishing \end{cases}$$

b. URL of Anchor

An element defined by the <a> tag is anchor. It is treated exactly as "Request URL". However, for this feature: Comparison of <a> tags and the website to check if they have different domain names, similar to the request URL feature.

$$R_{18}: URL = \begin{cases} if\ (\%of URL Of Anchor < 31\%) \rightarrow Legitimate URL \\ else if\left(\%of URL Of Anchor \geq 31\% And \leq 67\%\right) \rightarrow Suspicious \\ else \rightarrow Phishing URL \end{cases}$$

c. Abnormal URL

WHOIS database is used to extract this feature and identity typical part of URL for a legitimate website.

$$R_{19}: URL = \begin{cases} If\left(TheHostNameIsNotIncludedInURL\right) \rightarrow PhishingURL \\ else \rightarrow LegitimateURL \end{cases}$$

3) Content and blacklist base Rule

a. HTTPS

Presence of HTTPS is very important which gives the impression of website legitimacy, but this is clearly not enough. There are suggestion check the certificate assigned with HTTPS including the extent of the trust certificate issuer, and the certificate age. The list of Certificate Authorities that are consistently listed among the top trustworthy names include: "GeoTrust, GoDaddy, Network Solutions, Thawte, Comodo, Doster and VeriSign". The minimum age of a reputable certificate was found to be two years by checking out the datasets.

$$R_{20}: URL = \begin{cases} if\left(Use\,https\,and\,Issuer\,Is\,Trusted\,and\,Age\,of\,Cert. \geq 1Years\right) \rightarrow Legitimate \\ else\,if\left(\text{Using https and Issuer Is Not Trusted}\right) \rightarrow SuspiciousURL \\ else \rightarrow PhishingURL \end{cases}$$

b. Favicon

The graphic image (icon) associated with a specific webpage is favicon. Favicons are generally shown as a visual reminder of the website identity in the address bar by the existing user agents such as graphical browsers and newsreaders. Favicon loaded from a domain other than that shown in address bar is likely to be a Phishing attempt.

$$R_{21}: URL = \begin{cases} if\left(FaviconLoadedFromExternalDomain\right) \rightarrow Phishing \\ else \rightarrow Legitimate \end{cases}$$

c. The Existence of "HTTPS" Token in the Domain Part of the URL

The "HTTPS" token may be added to the domain part of a URL in order to trick users as in http://https-www-paypal-it-webapps-mpp-home.soft-hair.com/.

d. Having links in <Meta>, <Script> and <Link> tags

It is found from the webpage source code that legitimate websites offers metadata about the HTML document using <Meta> tags; creation of a client side script using <Script> tags; and to

retrieval of other web resources using <Link> tags. These tags should be linked to the same domain of the webpage.

$$R_{23}: URL = \begin{cases} if\ (\%ofLinksin\ "\langle Meta \rangle ",\ "\langle Script \rangle "and\ "<\text{Link}>"<17\%) \rightarrow LegitimateURL \\ (\%ofLinksin \langle Meta \rangle ",\ "\langle Script \rangle "and\ "<\text{Link}>" \geq 17\% And \leq 81\%\ ) \rightarrow Suspicious \\ else \rightarrow PhishingURL \end{cases}$$

e. Website Forwarding

The number of times a website has been redirected also distinguishes phishing websites from legitimate ones. It could be found from the dataset that legitimate websites have been redirected one time max, whereas the phishing websites have been redirected at least 4 times.

$$R_{24}: URL = \begin{cases} if\left(of\ Redirect\ Page \leq 1\right) \rightarrow LegitimateURL \\ else if(of\ Redirect\ Page \geq 2\ And < 4) \rightarrow SuspiciousURL \\ else \rightarrow PhishingURL \end{cases}$$

f. Status Bar Customization

JavaScript are used to show a fake URL in the status bar to users. This feature can be extracted by examining the webpage source code, particularly the "onMouseOver" event, to check if it makes any changes on the status bar.

$$R_{25}: URL = \begin{cases} If\left(onMouseOver\ Changes\ Status\ Bar\right) \rightarrow PhishingURL \\ It\ Does\ 't\ Change\ Status\ Bar \rightarrow LegitimateURL \end{cases}$$

g. By Disabling Right Click

JavaScript is used to disable the right-click function, so that the webpage source code cannot be viewed and saved. It is treated exactly as "Using onMouseOver to hide the Link". Also, "event. button==2" event is also searched for in the webpage source code to check if the right click is disabled.

h. Using Pop-up Window

Legitimate website will never ask users to submit personal information through a pop-up window. Pop-up windows are used by legitimate websites to warn users about fraudulent activities or broadcast a welcome announcement. No personal information is asked to be filled through these pop-up windows.

i. IFrame Redirection

An additional webpage can be displayed into the one that is currently shown by using IFrame tag in HTML. Attackers use the "iframe" tag to make these additional webpages invisible i.e. without frame borders. Also, frameBorder" attribute can also be used which renders a visual delineation in the browser

Xj Based on the number of Links Pointing to Page

In a webpage, legitimacy level of this is indicated by the number of links pointing towards itself even though several links has been pointing towards the same domain.

$$R_{29} : URL = \begin{cases} if\left(OfLinkPointingtoTheWebpage = 0\right) \rightarrow Phishing \\ elseif(OfLinkPointingtoTheWebpage > 0 and \leq 2) \rightarrow Suspicious \\ else \rightarrow Legitimate \end{cases}$$

k. Statistical-Reports Based Feature

Over a given span of time, numerous parties like Phish Tank, and StopBadware originated various statistical reports on spoofed sites. Most of them are carried out monthly and some others being quarterly. In the research, there are 2 forms, which show topmost 10 statistics from Phish Tank: "Top 10 Domains" and "Top 10 IPs" as stated in the statistical reports published from January 2010 to November 2012 over past three years where as in "StopBadware" uses "Top50"IP Addresses.

URL Matching Identity Feature Set

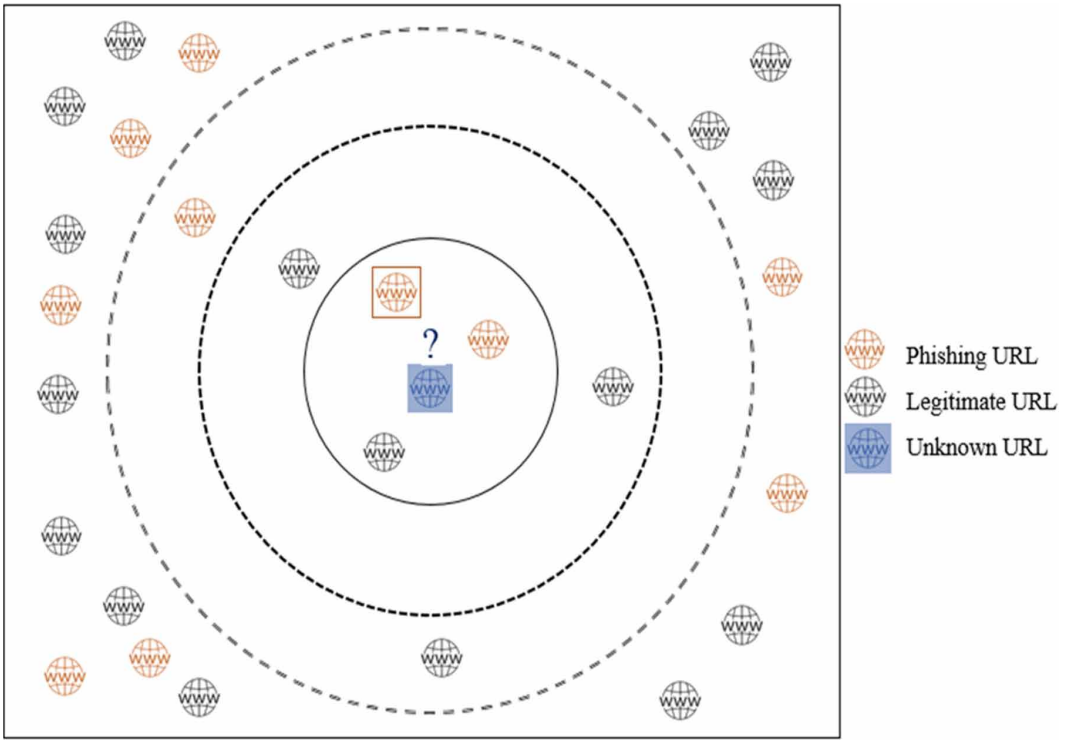
The idea behind k-Nearest Neighbor classifier is to check the aggregate score of similarity of a URL with the list of phishing URLs. In phishing attacks, the attacker tries to invite user to visit a fake webpage by different ways. In fact, he/she tries to create a false sense of confidence. One of these attraction ways is to register some addresses whose URL, at a glance, is similar to the address of the real website, so the novice user may not distinguish the difference. For example, the page address of https://www.0nlinesbi.com/is very similar tohttps://www.onlinesbi.com/but it is a phishing site of the "state bank of India" (At first URL, zero is re- placed as "O").

In such cases, direct comparison of two addresses may produce wrong result. For this reason, use approximate matching methods. Approximate string matching is a way for finding approximate patterns similar to a pattern in a textual string.

For that purpose, it has been suggested to use this estimated distance in a k-nearest neighbor (k-NN) scenario. The k-nearest neighbor classifier can be considered as one of the pioneers among the supervised methods – proposed originally by Fix and Hodges. Considering an annotated data collection, for an unknown data sample the class label is assigned based on the majority of its k-nearest neighbors. The so called nearest neighbor classifier is the special case of the previously mentioned one for k = 1. Even though it is a rather simple method, it has some indisputable advantages such as: simplicity, effectiveness, intuitively, non-parametric and high performance for different classification tasks. k-NN classification algorithm for phishing detection is shown in Figure 2.

1. Edit Distance

Figure 2. K-Nearest neighbor classification



It is a method of quantifying how dissimilar two strings are by counting the minimum number of operations required to transform one string into another. The edit distance between $X = (x_1, x_2 \ldots\ldots x_i)$ and $Y = (y_1, y_2 \ldots\ldots y_j)$ is given by $D_{r\,,}$ and is defined as:

$$R_{31} : D_{x,y}(i,j) = \begin{cases} Max(i,j)\ if\ \min(i,j) = 0 \\ Min\begin{cases} Dx,y(i-1,j)+1 \\ dx,y(i,j-1)+1 \\ Dx,y(i-1,j-1)+1(xi \neq yj) \end{cases} \quad otherwise \end{cases}$$

The value Dx,y shows the function of dissimilarity between X and Y. The similarity function is calculated by subtracting it from1 as shown in equation 2

$$sim(x,y) = 1 - Dx,y$$

EXPERIMENTAL SETUP

In this section, the detail information of the setup is provided. This includes how preparation of datasets and the metrics are used to evaluate the method.

Dataset

The process of identifying the type of a URL is generated using classification rules in which a different hybrid rule (URL matching, rule based) is utilized to acquire unknown knowledge. This rule is used to determine the URL type when a user accesses it. Brief legitimate data and phishing data source are shown in Table 2 and 3.

Table 2. Legitimate data source

S. NO.	Source	link
1	Yahoo most visited sites,(2015)	http://dir.yahoo.com/Business_and_Economy
2	Alexa's top targeted sites,(2015)	https://www.alexa.com/topsites
3	Stuffgate,(2015)	http://stuffgate.com/stuff/website/top-sites

Table 3. Phishing data source

S. NO.	Source	Link
1	PhishTank,(2015)	https://www.phishtank.com

Preprocessing

Since phishing sites are active for a short period, some challenges were faced during the process of data collection. For instance, the majority of phishing sites were infected with computer malwares, which cause some problems in the computer while data extracting. Most of phishing sites included in the PhishTank have not been classified properly, which led to decrease the speed of the data collection process. In order to prevent any further errors in datasets, preprocessing steps are used to prepare dataset for classification. In this step, some operations such as removing irrelevant data, eliminating the data with error and loss, and removing redundant data are prepared because of the impact of duplicate data in machine learning, sort the extracted features based on their webpage URL and then excluded duplicate data of a repetitive domain.

To scale all features value, mapped all data range into –1 to +1.

Rule Based Scoring

Firstly, the required features are extracted from the dataset. Each feature is individually analyzed. These features give the idea about the total number of Phishing and Real URL present in the given dataset. Next step is to arrange the dataset in order of the reference of the Phishing URL. After that percentage of module is calculated from the formula given below

$$\% \text{ of module} = \# \text{ of Phishing URL} / \text{ Total number of instanc e}$$

Brief collected rule based scoring are shown in Table 4. From the above equation, Percentile is calculated with reference to the highest percentage of phishing URL as shown in Table 5.

$$R_{\text{Percentile}} = \% \text{ of Phishing URL} / \text{ highest } \% \text{ of Phishing URL}$$

Table 4. Collected Rule based scoring

Rule	Real	Phishing	suspicious	%phishing	Rule	Real	Phishing	suspicious	%phishing
R_1	1106	6199	0	84.85969	R_{16}	6258	1047	0	14.33265
R_2	3144	4161	0	56.96099	R_{17}	4476	2829	0	38.7269
R_3	1283	5941	81	81.32786	R_{18}	1788	1803	3714	24.68172
R_4	6275	1030	0	14.09993	R_{19}	6165	1140	0	15.60575
R_5	3186	1830	2289	25.05133	R_{20}	4949	1688	668	23.10746
R_6	6259	1046	0	14.31896	R_{21}	6233	1072	0	14.67488
R_7	6118	1187	0	16.24914	R_{22}	6042	1263	0	17.28953
R_8	6636	669	0	9.158111	R_{23}	1769	2752	2784	37.67283
R_9	2322	4983	0	68.21355	R_{24}	957	147	6201	2.01232
R_{10}	3994	3311	0	45.32512	R_{25}	6626	679	0	9.295003
R_{11}	5762	1543	0	21.12252	R_{26}	7073	232	0	3.175907
R_{12}	4156	1560	1589	21.35524	R_{27}	6161	1144	0	15.66051
R_{13}	2321	4984	0	68.22724	R_{28}	6844	461	0	6.310746
R_{14}	6345	960	0	13.14168	R_{29}	2765	367	4173	5.023956
R_{15}	1302	5345	658	73.16906	R_{30}	6537	768	0	10.51335

Table 5. Collected phishing percentile after rule based scoring

Rule	Real	Phishing	suspicious	percentile	Rule	Real	Phishing	suspicious	percentile
R_1	1106	6199	0	10	R_{16}	6258	1047	0	1.6
R_2	3144	4161	0	6.7	R_{17}	4476	2829	0	4.5
R_3	1283	5941	81	9.5	R_{18}	1788	1803	3714	2.9
R_4	6275	1030	0	1.6	R_{19}	6165	1140	0	1.8
R_5	3186	1830	2289	2.9	R_{20}	4949	1688	668	2.7
R_6	6259	1046	0	1.6	R_{21}	6233	1072	0	1.7
R_7	6118	1187	0	1.9	R_{22}	6042	1263	0	2
R_8	6636	669	0	1	R_{23}	1769	2752	2784	4.4
R_9	2322	4983	0	8	R_{24}	957	147	6201	0.2
R_{10}	3994	3311	0	5.3	R_{25}	6626	679	0	1
R_{11}	5762	1543	0	2.4	R_{26}	7073	232	0	0.3
R_{12}	4156	1560	1589	2.5	R_{27}	6161	1144	0	1.8
R_{13}	2321	4984	0	8	R_{28}	6844	461	0	0.7
R_{14}	6345	960	0	1.5	R_{29}	2765	367	4173	0.5
R_{15}	1302	5345	658	8.6	R_{30}	6537	768	0	1.2

$$R_{scoring}\left(Rule based scoring\right) = R_{\text{Percentile}} / 10$$

$$R_{scoring}\left(Rule based scoring\right) = \frac{522820}{11055} = 47.3$$

After that, impact factor is calculated in which those feature are considered which are most effective. For example, enumerate prefix or suffix to the URL distinct by (-) hyphen, domain registration, and length of URL contain the highest impact factor. Impact factor of top three features is approximately 27.83%.

$$\left(10 + 8.0 + 9.5\right) * \frac{100}{98.8} = 27.83\%$$

Score is assign to all features with the help of percentile phishing value. By add the scores of each feature and divide it with total number of features for calculating the aggregate score. For the given dataset, it is around 3.2 and this score is further used for URL matching Identity method.

$$Aggregate Score = \frac{\sum_1^{30} R_i}{Total no. of rule\left(R\right)} \qquad Here 0 < i \leq 30$$

$$\frac{97}{30} = 3.2$$

URL Matching Identity Scoring

Firstly, in this feature set edit distance is calculated with the help of following formula

$$\mathrm{D}_{\mathrm{x,y}}\left(\mathrm{i,j}\right) = \begin{cases} Max\left(\mathrm{i,j}\right) if\ min\left(i,j\right)=0 \\ Min\begin{cases} Dx,y\left(i-1,j\right)+1 \\ dx,y\left(i,j-1\right)+1 \\ Dx,y\left(i-1,j-1\right)+1\left(xi \neq yj\right) \end{cases} \text{otherwise} \end{cases}$$

and this edit distance is used in k-NN classification algorithm.

For a given dataset instance x_q (new URL) is to be classified, Let $\mathrm{X} = (\mathrm{x}_1, \mathrm{x}_2 \ldots\ldots\ldots x_k)$ denote the k instances from training dataset that are nearest to x_q. Brief aggregate scoring is shown in table 7.

Table 6. Aggregate scoring

X	1	2	3	4	5	6	7
$\mathrm{X_q}$ (scoring)	3.2	6.4	9.6	12.8	16	19.2	22.4
% scoring	14.2	28.5	42.8	57.1	71.4	85.7	100

Here k=7 then in this case query distance x_q (New URL) will be classified as

$$x_q = \begin{cases} if\left(x \le 3\right) \rightarrow legitimate \\ elseif\left(x > 3\right) \rightarrow phishing \end{cases} \qquad Here\ 0 < x \le 7$$

Threshold

Threshold is calculated by adding the aggregate score of rule based feature set and the score of URL Matching Identity feature set.

$$T = x_{scoring} + R_{scoring}$$

In the given dataset, aggregate score using rule based feature set is 47.3.If aggregate score $R_{scoring}$ >47.3, then it is a phishing URL otherwise it is a legitimate URL. In the KNN dataset aggregate score using URL Matching Identity feature set is 9.6. If aggregate score $x_{scoring} > 9.6$, then it is a phishing URL otherwise it is a legitimate URL. Therefore threshold is given by

$$T = 47.3 + 9.6 = 56.9$$

If threshold T > 56.9, then it is a phishing URL otherwise it is a legitimate URL.

Feature Set

In proposed technique, there are 31 features to build the model. In order to identify the importance of the proposed features, it has been prepared three feature set as follow:

S1: Contains the URL matching identity based features.
S2: Contains the rule based features.
S3: All features include URL matching and rule based features (S1+S2), which results in total of 31 features

Training and Testing Data

In this a 10-fold cross-validation for computing the classification results are performed. The labeled dataset is divided into 10 subsets. In each test run, 9 subsets are used for training and the remaining subset is used as test data. In order to ensure that each set has been used for training as well as testing, 10-test run has been done. The final classification result is the aggregate of results from the 10 runs.

RESULTS AND MODEL EVALUATION

Numerous experiments are performed to evaluate the performance of proposed Phishshelter detection browser. The proposed system is compared with other popular and standard anti-phishing approaches. In order to signify the overall performance of proposed method, the experiment has been performed below:

Model Evaluation

The experiment has been performed using 10-fold cross validation to train and test the model. There are three features set design to build the model separately. Figure 6 displays the classification result. When the first feature set (S1) is used to classify rule based detection, the values of the above-mentioned characteristics (TPR, FNR) are 90.8% and 10.7%, respectively. Although, by using the URL matching feature set (S2) the values of them are 93.72% and 6.7%, respectively. The overall precision of the model was increased using the second feature set compared to the first too.

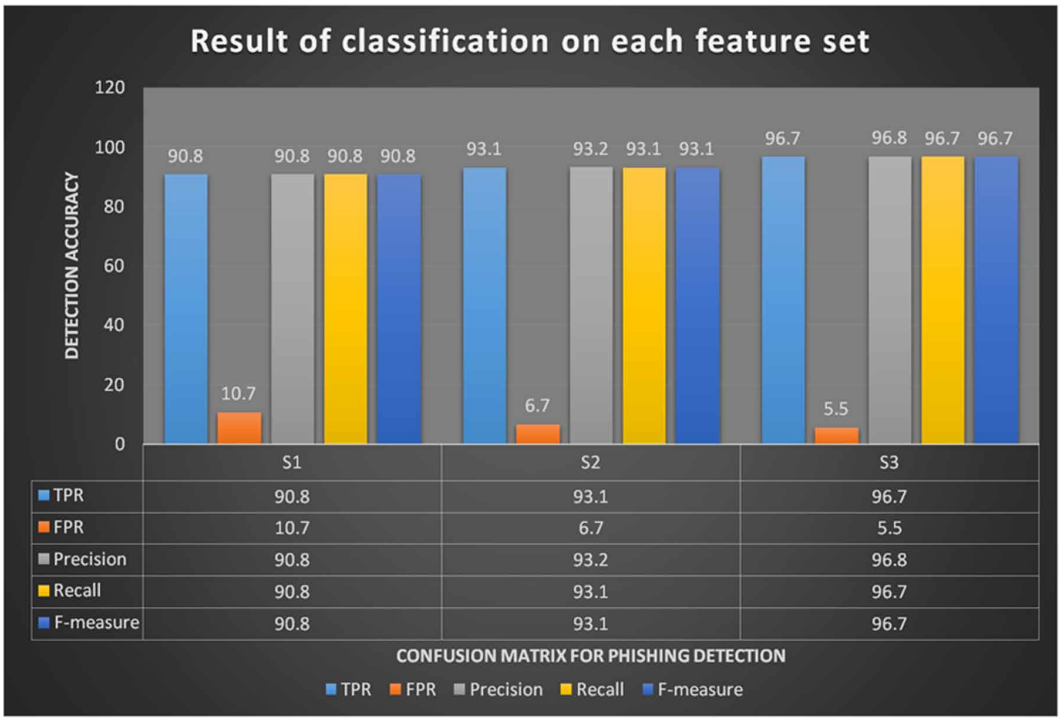
According to third feature set (S3), the accuracy of detecting phishing attack or the TPR characteristic increased to 96.7%. The value of FNR, which plays an important role in detecting semantic attack, also decreased to 5.5%. The fall indicates that the model demonstrates higher precision when a 31 total features to create the model and detect phishing pages in browser.

After decision table classification using the feature vector, which contains 31 total features in feature set (*S3*), it has been found that the proposed model is able to detect phishing pages in browser with accuracy of 96.70% and error rate of 6.5%. This error level is because of high volumes of errors linked with the FPR of the model in detecting legitimate pages as phishing. The acquired results from classification of web-pages and 31 features included in *S3* are shown in figure 3. The Kappa value is 0.928, which according to the Rule of Thumb indicates that the model has an acceptable output.

Comparison with Previous Phishing Detection Methods

In this section, the proposed method is compared with previous phishing detection methods. Table 7 shows the comparison which is based on True Positive Rate, False Positive Rate, language independent method, Search engine independent method. Although, the work of Patil and CANTINA has achieved slightly higher true positive rate but they are not search engines independent. Similarly, the work of

Figure 3. Result of classification on each feature set



Zhang has achieved higher true positive rate however, it is Language Independent. The proposed work is language as well as search engine independent with reasonable true positive rate.

Evaluating of Proposed Feature Sets

To calculate the effect of each regarding feature on output of the classification, an employed sensitivity analysis. In sensitivity analysis, the variability of outputs changes is measured through the variability of inputs changes. In proposed technique we have used, one-at-a-time method to evaluate the effect of each feature of the feature vector. In this method, for changing the entries of each category, model output statistics is measured. Finally, according to the sensitivity of classification model, the effectiveness of each desired feature is measured. An accuracy, error, kappa, sensitivity, and F-Score are used to measure each feature importance in sensitivity analysis. Table 8 details the model statistics in eliminating of each feature.

Overall, the experiments are demonstrated that the proposed extension can detect phishing web-pages with 96.7% accuracy. In this, proposed model with the lowest rules number produced higher true positive rate shows in table 9.

CONCLUSION

With the growing awareness of technology in all aspect of human life, attackers always try to use new methods to target their victims too. It is necessary to improve methods and techniques to detect these frauds, and prevent more financial losses.

The proposed technique primarily focused on the phishing attacks which are performed through URL and proposed method work on these attacks. In order to improve false positives and negatives on newer data sets, work should be done on refining the rules.

This research discloses many challenges in field of web application security. The experimental results showed that the proposed approach is very effective in protecting against phishing attacks as it has 96.7% true positive rate with a very less false positive rate of 5.5%. In addition, proposed system is efficient to detect various other types of phishing attacks (i.e., DNS poisoning, embedded objects, zero-hour attack). Moreover, this approach is suitable for a real-time environment. In the future, the performance of the proposed system can be improved by taking the other features along with the hyperlinks; however, extracting other features will increase the running time complexity of the system.

Table 7. Comparison with previous phishing detection methods

Work	Method	True Positive Rate	False Positive Rate	Language Independent	Search engines Independent
Zhang et al., 2007	CANTINA	97	6	No	No
Hossain et al., 2010	Fuzzy logic	84.2	15.7	Yes	Yes
Ramesh et al., 2014	New approach	99	0.9	yes	No
Zhang et al., 2014	New method	98	0.53	No	Yes
Rao & Pais, 2018	Machine Learning (SVM)	94.38	11.07	Yes	Yes
Peng et al., 2018	Natural Language Processing	95	9	No	Yes
Patil et al., 2019	Hybrid Method	99	5	Yes	No
Proposed work	**Hybrid**	**96.7**	**5.5**	**Yes**	**Yes**

Table 8. Result of sensitivity analysis

Feature	Accuracy	Error	Kappa Sensitivity	F1-measure	Feature	Accuracy	Error	Kappa Sensitivity	F1-measure
R_1	83.8	0.24	0.66	0.8	R_{17}	95.7	0.1	0.82	0.95
R_2	84.3	0.18	0.34	0.81	R_{18}	97.1	0.06	0.89	0.97
R_3	98	0.059	0.91	0.98	R_{19}	92.7	0.15	0.6	0.91
R_4	92.3	0.14	0.67	0.91	R_{20}	98	0.07	0.87	0.98
R_5	98.2	0.05	0.92	0.98	R_{21}	99	0.08	0.83	0.99
R_6	84.9	0.25	0.1	0.77	R_{22}	98.8	0.03	0.95	0.98
R_7	59.5	0.3	0.36	0.59	R_{23}	97.5	0.07	0.76	0.97
R_8	86.4	0.14	0.72	0.85	R_{24}	76	0.32	0.51	0.75
R_9	83.9	0.27	0.62	0.83	R_{25}	93	0.12	0.79	0.93
R_{10}	99.1	0.03	0.96	0.99	R_{26}	64	0.33	0.32	0.61
R_{11}	99	0.03	0.93	0.99	R_{27}	75	0.34	0.37	0.73
R_{12}	94.8	0.1	0.81	0.94	R_{28}	89.5	0.19	0.35	0.87
R_{13}	83.2	0.27	0.63	0.82	R_{29}	81.6	0.2	0.64	0.81
R_{14}	69.8	0.29	0.49	0.82	R_{30}	90.1	0.17	0.29	0.88
R_{15}	63.3	0.34	0.43	0.63	R_{31}	98.9	0.08	0.79	0.98
R_{16}	78.8	0.27	0.37	0.75					

Table 9. Confusion Matrix

Kappa Statistics	Time Taken To Build Model(SEC.)	F-Measure	MCC	ROC Area	Precision	Recall
0.928	2.33	0.967	0.93	0.976	0.968	0.967

REFERENCES

Aburrous, M., Hossain, M. A., Dahal, K., & Thabtah, F. (2010, April). Predicting phishing websites using classification mining techniques with experimental case studies. In*2010 Seventh International Conference on Information Technology: New Generations*(pp. 176-181). IEEE. doi: doi:10.1109/ITNG.2010.117

Alexa's top targeted sites. (2016). Retrieved fromhttp://www.alexa.com/topsites

Anti-Phishing Working Group. (2017). Retrieved fromhttps://docs.apwg.org/

Aravindhan, R., Shanmugalakshmi, R., Ramya, K., & Selvan, C. (2016, January). Certain investigation on web application security: Phishing detection and phishing target discovery. In*2016 3rd International Conference on Advanced Computing and Communication Systems (ICACCS)*(Vol. 1, pp. 1-10). IEEE.

Cendrowska, J. (1987). PRISM: An algorithm for inducing modular rules. *International Journal of Man-Machine Studies*, *27*(4), 349–370. doi:10.1016/S0020-7373(87)80003-2

Find Website Traffic, Statistics, and Analytics. (2017). Retrieved from https://www.alexa.com/siteinfo

Google Webmaster. (2017). Retrieved fromhttps://www.alexa.com/siteinfo

Hara, M., Yamada, A., & Miyake, Y. (2009, March). Visual similarity-based phishing detection without victim site information. In *Computational Intelligence in Cyber Security, 2009. CICS'09. IEEE Symposium on* (pp. 30-36). IEEE. doi: doi:10.1109/CICYBS.2009.4925087

Heartfield, R., Loukas, G., & Gan, D. (2017, June). An eye for deception: A case study in utilizing the human-as-a-security-sensor paradigm to detect zero-day semantic social engineering attacks. In*2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)*(pp. 371-378). IEEE. doi: doi:10.1109/SERA.2017.7965754

Joshi, Y., Saklikar, S., Das, D., & Saha, S. (2008, December). Phishguard: a browser plug-in for protection from phishing. In*2008 2nd International Conference on Internet Multimedia Services Architecture and Applications*(pp. 1-6). IEEE. doi: doi:10.1109/IMSAA.2008.4753929

Ma, B. L. W. H. Y., & Liu, W. (1998, August). Integrating classification and association rule mining. *Proceedings of the fourth international conference on knowledge discovery and data mining*.

Mao, J., Tian, W., Li, P., Wei, T., & Liang, Z. (2017). Phishing-alarm: Robust and efficient phishing detection via page component similarity. *IEEE Access : Practical Innovations, Open Solutions*, *5*, 17020–17030. doi:10.1109/ACCESS.2017.2743528

Nagunwa, T., Naqvi, S., Fouad, S., & Shah, H. (2019, May). A Framework of New Hybrid Features for Intelligent Detection of Zero Hour Phishing Websites. In*International Joint Conference: 12th International Conference on Computational Intelligence in Security for Information Systems (CISIS 2019) and 10th International Conference on EUropean Transnational Education (ICEUTE 2019)*(pp. 36-46). Springer.

Patil, V., Thakkar, P., Shah, C., Bhat, T., & Godse, S. P. (2019, April). Detection and Prevention of Phishing Websites Using Machine Learning Approach. In*2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*(pp. 1-5). IEEE.

Peng, T., Harris, I., & Sawa, Y. (2018, January). Detecting phishing attacks using natural language processing and machine learning. In*2018 IEEE 12th International Conference on Semantic Computing (ICSC)*(pp. 300-301). IEEE. doi: doi:10.1109/ICSC.2018.00056

Philippsohn, S. (2001). Special Features: Trends in Cybercrime-An Overview Of Current Financial Crimes On The Internet. *Computers & Security*, *20*(1), 53–69. doi:10.1016/S0167-4048(01)01021-5

Rahman, R. U., & Tomar, D. S. (2018). Botnet Threats to E-Commerce Web Applications and Their Detection. In *Improving E-Commerce Web Applications Through Business Intelligence Techniques* (pp. 48–81). IGI Global.

Ramesh, G., Krishnamurthi, I., & Kumar, K. S. S. (2014). An efficacious method for detecting phishing webpages through target domain identification. *Decision Support Systems*, *61*, 12–22. doi:10.1016/j.dss.2014.01.002

Rao, R. S., & Pais, A. R. (2018). Detection of phishing websites using an efficient feature-based machine learning framework. *Neural Computing & Applications*, ●●●, 1–23.

Receiver operating characteristic. (2017). Retrieved fromhttps://en.wikipedia.org/wiki/Receiver_operating_characteristic

Sensitivity Analysis. (2017). Retrieved fromhttp://www.investopedia.com/terms/s/sensitivityanalysis.asp

Sheng, S., Wardman, B., Warner, G., Cranor, L., Hong, J., & Zhang, C. (2009). *An empirical analysis of phishing blacklists*. Academic Press.

Stuffgate. (2016). Retrieved from http://stuffgate.com/stuff/website/top-sites

Sumner, A., & Yuan, X. (2019, April). Mitigating Phishing Attacks: An Overview. In *Proceedings of the 2019 ACM Southeast Conference* (pp. 72-77). ACM. doi: doi:10.1145/3299815.3314437

Thabtah, F., Cowling, P., & Peng, Y. (2005, January). MCAR: multi-class classification based on association rule. In*The 3rd ACS/IEEE International Conference onComputer Systems and Applications, 2005*(p. 33). IEEE. doi: doi:10.1109/AICCSA.2005.1387030

Unifying the Global Response to Cybercrime. (2012). Retrieved fromhttp://www.antiphishing.org/

ur Rahman, R., Tomar, D. S., & Das, S. (2012, May). Dynamic image based captcha. In *Communication Systems and Network Technologies (CSNT), 2012 International Conference on* (pp. 90-94). IEEE.

ur Rizwan, R. (2012). Survey on captcha systems. *Journal of Global Research in Computer Science, 3*(6), 54-58.

Whois database. (2017). Retrieved from https://who.is/

Wu, M., Miller, R. C., & Garfinkel, S. L. (2006, April). Do security toolbars actually prevent phishing attacks? In *Proceedings of the SIGCHI conference on Human Factors in computing systems* (pp. 601-610). ACM.

Yahoo most visited sites. (2016). Retrieved fromhttp://dir.yahoo.com/Business_and_Economy

Zhang, Y., Hong, J. I., & Cranor, L. F. (2007, May). Cantina: A content-based approach to detecting phishing web sites. In*Proceedings of the 16th international conference on World Wide Web*(pp. 639-648). ACM.

Rizwan ur Rehman is currently working as Assistant Professor in the Department of Computer Science and Engineering/Information Technology, Jaypee University of Information Technology, Waknaghat, Solan, Himachal Pradesh. Previously, he worked as Assistant Professor (on contract) in the Department of Computer Science and Engineering, NIT Bhopal. He has over 8 years of teaching experience. His programming experience includes C/C++, C#, SQL, PHP, ASP, ASP.NET, VB, VB.NET; Win Forms, Web Forms and Java. He has worked on government projects and R&D department of CRISP.

Lokesh Yadav received the Bachelor of Technology (B.Tech.) degree in Computer Science and Engineering from National Institute of Technology (NIT), Agartala, India in 2014 and Master of Technology (M.Tech.) from the Department of Computer Science and Engineering, Maulana Azad National Institute of Technology (MANIT), Bhopal, India in 2017. He is currently a Ph.D. Scholar in the Department of Computer Science and Engineering, Maulana Azad National Institute of Technology (MANIT), Bhopal, India. His research interests include cybersecurity, machine learning, and data mining.

Deepak Singh Tomar obtained his B.E., M.Tech., and Ph.D. degrees in Computer Science and Engineering. He is currently Assistant Professor of CSE department at NIT- Bhopal, India. He is co-investigator of Information Security Education Awareness (ISEA) project under Govt. of India. Currently, he is chairman of cyber security center, MANIT, Bhopal. He has more than 21 years of teaching experience. He has guided 30 M Tech and 3 PhD Thesis. Besides this he guided 70 B Tech and 15 MCA projects. He has published more than 54 papers in national & international journals and conferences. He is holding positions in many world renowned professional bodies. His present research interests include web mining and cyber security.