Naif Arab University for Security Sciences
Journal of Information Security & Cybercrimes Research
مجلة بحوث أمن المعلومات والجرائم السيبرانية
https://journals.nauss.edu.sa/index.php/JISCR
JISCR

# A URL with Image-based Feature Extraction for Preventing Phishing Attacks

**Dyaa Eldeen Nasr Motawa [1], Ahamed El Shrief [2]***
[1] *Bachelor of Computer Science, Arab Open University*
[2] *Department of Information Security, College of Computer and Information Security, Naif Arab University for Security Sciences, Riyadh, Saudi Arabia.*

*Received 01 Mar. 2019; Accepted 15 May. 2019; Available Online 05 Jun. 2019*

## Abstract

Currently, Phishing is a type of attack in which cyber criminals tricks the victims to steal their personal and financial data. It has become an organized criminal activity. Spoofed emails claiming to be from legitimate source are crafted in a way to lead victims to reveal their personal, financial data by misdirecting them to the counterfeit website. We compared previous password protection solutions, some of the presented solutions use specialized equipment or additional servers to protect passwords. Other solutions are prone to spoofing and phishing attacks as well as introduce usability issues. Also these solutions do not address the challenge of protecting passwords against the adversary who can, for instance, exploit server-side software vulnerabilities. Our goal is enhancing the best solution to prevent phishing by alerting the users from phishing websites when detected based on URL with image-based feature extraction method.

## I. Introduction

The significance of network attacks is increasing day by day as the size and sensitivity of data being transferred across the Internet increase. Phishing attacks are not limited to spoofed emails only; it includes search engines, man-in-middle, malware, Trojans, instant messaging, social networking sites and etc. Criminals also create fake web sites that masquerade as trustworthy organizations to disclose user's sensitive information. This issue motivate the researchers to do many studies to provide the needed security. As a solution for this issue we presents a creative URL with image-based feature extraction method for phishing site detection. Our technique works by first taking a screen capture of an objective page, at that point finding "visual hotspots" in it. A visual hotspot is a ceaseless rectangular district that contains non-text visual data. These hotspots represent image features of the target webpage. The features are then compared with the pre-built library. If any of these features match a logo in the logo library, the webpage is recognized as a phishing website.

The rest of this paper is structured as follows: Section 2 discusses the background. Our methodology is presented in Section 3 followed by Section 4, which evaluates the design and implementation in the form of a user study. Finally, Section 5 presents overall conclusions and suggestions for future work.

## II. Background

### *A. Passwords*

In the mid-1960s the Massachusetts Institute of Technology decided to utilize passwords authentication method so that different researchers could use the same

**Keywords:** *Phishing attacks, Feature extraction, Image processing, Speedup.*

Production and hosting by NAUSS

* Corresponding Author: Ahamed El Shrief
Email: aelsherif@nauss.edu.sa

doi: 10.26735/16587790.2019.006

computer without having access to the resources of others. Passwords have rapidly become the standard authentication mechanism on the web and this will likely not change over the next few years [1], [2]. The reasons for this are [3]:

1. Passwords are cheap and easy to use, and they do not require any specialized hardware in comparison to biometrics methods [5].
2. Passwords are easy to remember [4] and well understood by users.

**However, these characteristics also introduce some weaknesses of this authentication method:**

1. Users usually choose passwords that are relatively short and easy to guess [4].
2. Users reuse passwords across multiple websites [7] [6].
3. Passwords can be captured (eg. using sniffers) and used to impersonate the user.

*B. Related works*

Many studies have tended to the issue of phishing in mean time. Every of these studies approaches the issue of phishing with a different method. Subsequently, key components of every method are investigated in the rest of the pieces of this section.

The researchers Jain and Gupta [8] proposed a model utilizing auto-update whitelist of legitimate sites and caution the clients if the URL is inaccessible in the whitelist. They confirm the authenticity of a webpage dependent on two segments: 1) Domain and IP address coordinating module, 2) Examine the features of the hyperlinks from source code. The outcome from the investigation demonstrates that the proposed model assessed 86.02 % true positive rates, furthermore, 1.48 % false negative rates.

Another study by He et al [9] has displayed a heuristic model by choosing 12 features from existing legitimate and phishing pages. Among these 12 features, nine features were received from the Anomaly Method, two features from appropriate method, and the one include from CANTINA method. Subsequently, they utilized Support Vector Machine (SVM) to classify the phishing and legitimate site with these 12 features. The results delivered to uncover that it could join diverse methods to improve the locator execution since the joined methods are correlative to existing methods.

The researchers Pan and Ding, in 2006 [10] analyzed the oddities in web pages, particularly, the divergence between a website's personality, its auxiliary features, and HTTP exchanges. The proposed method has two segments: 1) Identity extractor: the personality is a truncation of the association's full name and an extraordinary string showing up in its area name, 2) Page classifier bargains with two source structure features; one is W3C DOM protests on a web page, and another is HTTP exchanges. Finally, The researchers completed the examinations on 279 phishing pages and 100 authority pages utilizing support vector machine, which assessed false positive rate estimated 13.00%.

Another study by Islam and Abawajy [11] has proposed a multi-level classification model for phishing email sifting dependent on a weighting of message substance and message header and chooses the features as per the priority ranking. The outcome from the tests demonstrates that the proposed calculation decreased the false positive issues generously with lower complexity.

The features of URL, for example, transport layer security, inaccessibility of the top dimension space in the URL and catchphrase inside the way segment of the URL has been used in another study by Jeeva and Rajsingh [12]. Likewise, a few slashes in the URL, dot in the host segment of the URL and the length of the URL are additionally the key components for phishing URL. At that point, they produced a rule using association rule mining. The outcome from the study demonstrates that the apriori calculation distinguished rough 93.00% of the phishing URL.

A phishing webpage detection approach dependent on semi-managed learning method named Transductive Support Vector Machine (TSVM) has likewise been endeavored Li et al [13]. They removed the features from the web picture and delicate data on the page, and they characterized the phishing webpage utilizing TSVM. From the consequence of the investigation, the creators educated that the proposed model performs well and improved the exactness of 8.3% in study with support vector machine.

The page signature was created utilizing term recurrence has been embraced by another study [14]. This signature was sustained to a web index to recognize the genuine page and the resultant pages were analyzed utilizing label correlation and cosine comparability. The outcome got through this method demonstrates that the false positive rate was nearly low in the proposed instrument as it uses the Google page ranking data and identification rate was just as high contrasted and other existing systems.

String coordinating method is additionally utilized in recognizing phishing pages. Two string coordinating calculations, that is, Longest Common Subsequence

(LCS) and Edit Distance were used in an study by Abraham and Raj [15] to identify phishing assaults. The URL was separated into an alternate number of tokens, and the scores were processed dependent on the quantity of the event of every token in the boycott. The outcome from the investigation demonstrates exactness rates 99.1% and 99.5% for LCS and Edit Distance separately.

A few URL features like Host with an IP address, Host with another Domain, huge host names and space obscure or incorrectly spelled were utilized in study to recognize phishing by Garera et al [16]. They did the investigation utilizing strategic relapse channel and accomplished an exactness of 97.3%.

A model titled PhishDef has been created to distinguish phishing sites utilizing URL names by Le et al [17]. The creators investigated three stages to distinguish the phishing URL: 1) select the lexical features of URLs, 2) look at the precision of just lexical features, both consequently and hand-chose, versus extra features, 3) an online method (AROW) was proposed dependent on the study a few classification calculations. The outcome from this trial demonstrates that the lexical features were adequate for every single functional reason, and the proposed method accomplished an exactness of 97%.

Another study has implemented phishing identification by taking the screen capture of the specific area of the webpage which contain the logo of the web page, and after that they feed the logo to the Google Image Search engine to distinguish the webpage Chang et al [18]. With recovery of the genuine character, the creators recognized the phishing site with the legitimate site. The experiments show promising outcomes, and their discoveries demonstrate that it could viably recognize a phishing website by figuring out how to decide the genuine character of a website.

The Authors Alkhozae and Batarfi [19] proposed a model where they separated phishing described from websites dependent on the W3C standard. After correlation of both the pages of phishing sites and legitimate sites, they presumed that phishing sites were the less security rate than legitimate sites. An anti-phishing procedure dependent on a weighted URL token framework has likewise been created by scientists Tan et al. [20]. They extricated personality catchphrases from a question webpage and utilized these character watchwords as the mark of the page which was encouraged into an internet searcher to pinpoint the objective space name. They completed the tests with standard datasets, where 99.20 % true positives and 92.20 % true negatives were accomplished. The outcomes from the trial recommended that the proposed framework distinguished phishing webpages adequately without utilizing traditional language dependent keyword extraction calculations.

Another examination by Basnet and Doleck [21] has assembled lexical based, catchphrase based, web search tool based, and notoriety based features. Generally speaking with 138 features have been utilized to recognize phishing URLs. They finished their investigation on various roads with respect to 7 distinctive AI classifiers and found that arbitrary backwoods classifier surveyed prevalent and Naive Bayes evaluated most recognizably terrible execution. They acquired 0.2% false positive and 0.5% false negative rates.

Another model titled by Sonowal and Kuppusamy [22] MASPHID has been created by analysts, which helped screen per user clients in recognizing phishing sites using aural and visual likeliness measures. The model keeps up a whitelist of surely understood banks and confirms the present URL in the whitelist. In the event that the ebb and flow URL is missing in whitelist, at that point takes the screen capture of the ebb and flow site and concentrates the top dimension area of the ebb and flow URL and feed into the web crawler. The model chooses the main outcome from the internet searcher result and takes a screen capture and analyzes the two pages utilizing the root mean square method. In the event that the score of the root-mean-square mistake method is not exactly the edge esteem, at that point caution clients to visit this site. The outcome from the examination demonstrates that the aural and visual estimates were a viable strategy to identify phishing site for people with visual hindrances. In any case, this model applies to just people with visual weaknesses.

Heuristic-based phishing identification procedure that utilizes the uniform asset locator (URL) features has been proposed by Lee et al in 2015 [23]. They gathered features by Google's proposal, page ranking, suspicious URL examples and URL property estimations, and two novel features were joined for recognizing phishing URLs. The produced classifiers through several AI calculations and discovered that best classifier method is irregular woods method. The outcome from the trial demonstrates the proposed method recognized phishing site around 98.23%. In spite of the fact that, the creator proposed a helpful system, yet their features were excessively.

Another examination by Dhinakaran et al in 2010 [24] proposed a way to deal with recognize phishing assaults utilizing a multilayer approach that is a mix tweaking of spamming systems, hinder the aggressor's IP addresses, utilizing source and substance channels to check phishing endeavor, teach and train clients, report to

specialist co-ops, and target organizations. The proposed methodology could deal with inexact 95% of phishing assaults in their system.

The proposed PhiDMA model has received a multifilter approach. The falling of channels received as the model encourages separating with the help of features in different measurements. The availability include proposed by the PhiDMA model is a novel exertion to use the openness score of web pages in finding the similitude.

## III. Methodology

### *A. Research methodology*

Firstly, our design of the system has to allow all the components to operate without noticeably affecting the performance on the web. Secondly, sufficient level of security has to be provided by the system and it has to be understandable for users. All of the design decisions were made based on these objectives. Our anti phishing client-side browser extension involves two phases training and testing phase. In training phase we focus on the extraction of ”features”, or notable visual elements from a webpage which is finding "visual hotspots" in it. A visual hotspot is a ceaseless rectangular district that contains non-text visual data. These hotspots represent image features of the target webpage, see Fig. 1 for more details.

Our goal is to isolate recognizable logos and save it in the database in the testing phase. We observe that most phishing websites, will at least include a logo. Thus this provides us a perfect target for phishing detection.

To extract image features from the website in the training and testing phase we take:

1. The value of the corresponding source address of the image.
2. grayscale: we convert color array input [r0,g0,b0,a0, ...] to grayscale using Y = 0.299*R + 0.587*G + 0.114*B formula. We can specify the source input channel order such as BGRA, RGBA, RGB and BGR.
3. Features 2D extraction: we detects corners using the FAST algorithm.

Final code to extract features is listed in Appendix A.

Screenshots are processed locally in the browser and discarded immediately; they are not stored or sent to an external site.

In testing phase given the image features extracted from a suspicious webpage, we want to know whether they are identical or not to known logos we have in the library.

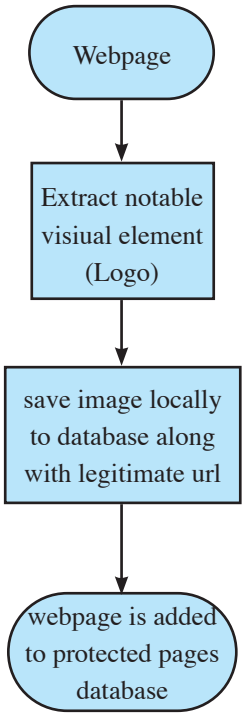


Fig. 1. Training phase.

We also added a safe domain detection method which is based on predefined domains that is secure and highly unlikely to host phishing webpages. These include Google, Amazon, Facebook and every domain for which we have added to safe domain list. We skip enhanced mode checking for these pages to reduce the CPU overhead. In testing phase given the image features extracted from a suspicious webpage, we want to know whether they are identical or not to known logos we have in the library as stated in Fig. 2. The first filter checks a given Web page for a login field. The second filter compare the image features extracted from a suspicious webpage, we want to know whether they are identical to known logos. After login form is detected a screenshots of the active browser tab are taken and compared with protected pages stored in database. The user is alerted if the current page visually imitate a protected page, but belongs to an unknown domain.

### *B. Client-side browser extension*

In our browser extension we propose a whitelist with an image feature previously taken technique to verify each page, which creates a huge perceptual difference between a real login page and a fake (phishing) page, such that even the users who are browsing and they have a weak knowledge and no awareness of phishing

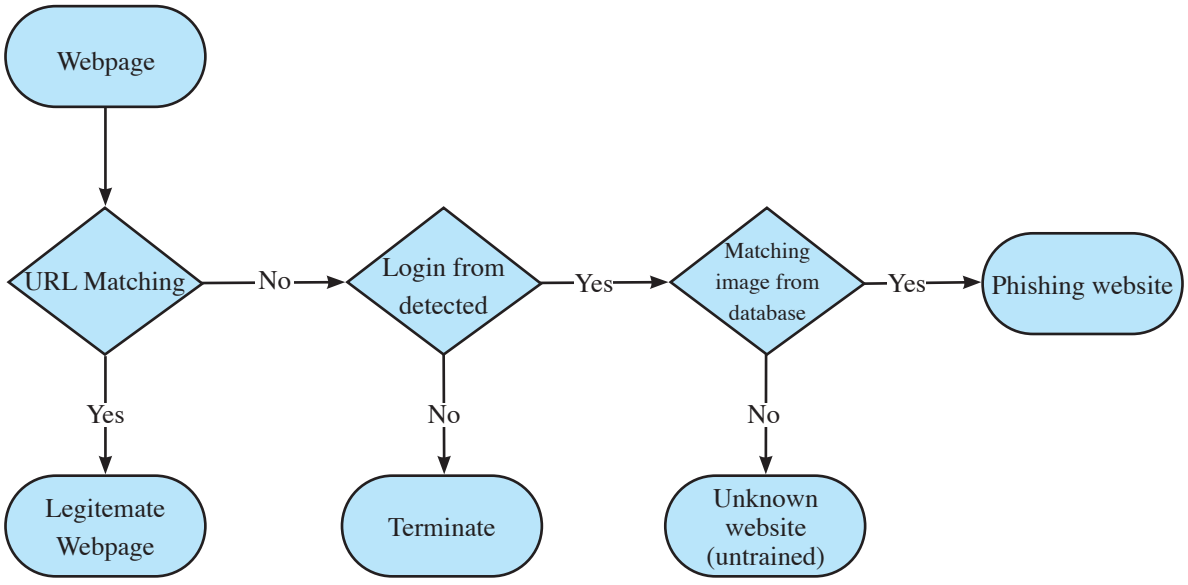


Fig. 2. sting phase.

attacks can instantly tell that something is wrong when they navigate to a phishing page. Which can be added to existing web browsers e.g. by installing a browser addon. The browser extension is designed for the Google Chrome web browser because Google provides detailed documentation as well as many useful APIs to build such extensions. However, similar implementations can be developed for the other web browsers (e.g. Firefox). This extension consists of two main parts:

1. The background script, which is responsible for all background processes such as processing webpage URL, login form detection and web requests, which happen within the extension.
2. The content script, which is injected to the website; therefore, this is the component that interacts with the website's content.

**Login form detection**

Almost all phishing attacks lure users into giving their information through a fake login form that looks like the legitimate login form. We detect login form to speed up runtime performance.

**Informing the user**

In our design, the extension has 4 states, Fig. 3:

1. Disabled.
2. Blue-highlighted.
3. Red-highlighted.
4. Green-highlighted.

State disabled if there is no any website to test. State blue-highlighted if we don't have any extracted feature

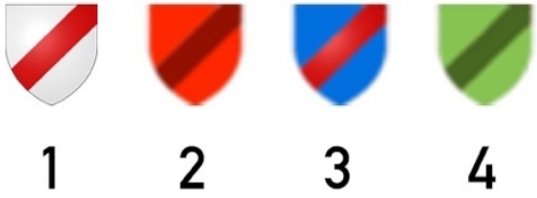


Fig. 3. The states of the extension.

about the webpage and it looks clean. That means the web server does not use the password protection service. State red-highlighted if the website is recognised as a phishing website. To fire the red-highlighting of suspect pages, we need to recognize whether the page loaded on the current active browser tab is visually similar to the login pages of protected sites. The extension check all image-like elements (divs with backgrounds, img elements and svg elements) against a list of images snippets of each protected site. Phishers tend to copy original (real) pages as far as possible, so this approach would work in a majority of cases. To summarize how the phishing detection work is that we take a screenshot of the active tab when the page is loaded, which gives us an image of the rendered page as the user saw it. Then we search the screenshot for the presence of image snippets from the login page of every protected site. Finally, Green-highlighted state indicates that the server utilizes

password protection service, however, the protected input fields are not highlighted. The Highlighted state appears when the user clicks the icon to display protected input fields.

## IV. Performance Analysis and Evaluation

In this chapter we assesses the security, performance, and usability aspects of our anti phishing protection system. In addition‘ the utility and usability of our protection system browser extension are assessed based on user study results. Various measurements were performed to evaluate the performance (ire. scalability and latency) of our protection system.

### *A. Performance analysis of our image-based feature extraction*

Our anti phishing browser extension had a delay of approximately 1824 ms when trying to detect Facebook phishing website and had a delay of approximately 1425 ms. The results are presented in Tables I.

TABLE I
Execution Delay

| | Phishing website | Time to detect |
|---|---|---|
| Test 1 | http://facbook.site | 1824 ms |
| Test 2 | https://diagroovy.com/nauss/ | 1452 ms |
| Test 3 | https://diagroovy.com/facebok/ | 1859 ms |
| Avg. | | 1711 ms |

### *B. User study*

In addition to complete the security requirements, the usability and utility of our browser extension were assessed by user study participants. This user study was conducted to verify if the system is usable and easy to understand.

#### *1) Method*

**Participants**

56 participants were chosen from the age group of 16 to 52. Table II shows how many participants have obtained different degrees. Nearly 58% of the testers do not usually check for the secure connection while browsing the web. 75% of the participants are aware of phishing.

After 2 weeks we invited 10 of the original participants to participate in a follow up user study without the extension to see if they could identify phishing sites. These participants were selected based on their scores from initial user study.

**Materials**

The following were given to the participants during the user study:

1. Video tutorial on how to setup the extension.
2. The browser extension.
3. Google form questionnaire.

All the forms are included in Appendix A. The information sheet states the purpose of the research project as well as informs who is in charge of the user study. This document also describes the testing procedure, safety guarantees, rewards and a voluntary nature of the experiment The consent form states that in order to participate in the experiment the participant has to be aware and agree to all the conditions described in the google form. The main questionnaire consists of brief instructions, the question, which applies to all 15 websites, as well as a links of these 15 websites. Using this links, the participants indicate their answer to the question separately for each website and they are also asked to indicate the level of certainty for each answer.

**Design and procedure**

The user study consisted of the following steps:

1. Tester receives the google form to sign.
2. Tester receives the questionnaire with 15 websites which also contains the question and brief instructions on how the test is conducted.
3. Test leader briefly emphasizes the most important information from the questionnaire and responds to any questions.
4. Tester reads the instructions for using the extension. These instructions were provided in the form the user would see when installing the browser extension.
5. Tester fills out the form while visiting the websites on their laptops. Testers do not have to enter any passwords. but only specify which fields on the form are protected.
6. Tester hands over the completed form.

Testers had to visit 15 web pages and they were asked to decide which websites use our protection system to protect passwords and which websites do not. The

TABLE II
PROBABILITY VALUES CALCULATED FOR PARTICIPANTS' OVERALL SCORES

| Participant number | Overall score out of 15 websites | Overall score (%) | p-value |
|---|---|---|---|
| 1 | 11 | 73% | 0.007 |
| 2 | 15 | 100% | <0.001 |
| 3 | 15 | 100% | <0.001 |
| 4 | 14 | 93% | <0.001 |
| 5 | 13 | 86% | <0.001 |
| 6 | 14 | 93% | <0.001 |
| 7 | 14 | 93% | <0.001 |
| 8 | 15 | 100% | <0.001 |
| 9 | 15 | 100% | <0.001 |
| 10 | 15 | 100% | <0.001 |
| 11 | 15 | 100% | <0.001 |
| 12 | 14 | 93% | <0.001 |
| 13 | 9 | 60% | 0.054 |
| 14 | 15 | 100% | <0.001 |
| 15 | 11 | 73% | 0.007 |
| 16 | 14 | 93% | <0.001 |
| 17 | 15 | 100% | <0.001 |
| 18 | 15 | 100% | <0.001 |
| 19 | 15 | 100% | <0.001 |
| 20 | 15 | 100% | <0.001 |
| 21 | 15 | 100% | <0.001 |
| 22 | 15 | 100% | <0.001 |
| 23 | 15 | 100% | <0.001 |
| 24 | 15 | 100% | <0.001 |
| 25 | 15 | 100% | <0.001 |
| 26 | 15 | 100% | <0.001 |
| 27 | 15 | 100% | <0.001 |
| 28 | 15 | 100% | <0.001 |

TABLE II
PROBABILITY VALUES CALCULATED FOR PARTICIPANTS' OVERALL SCORES *(Continued.)*

| Participant number | Overall score out of 15 websites | Overall score (%) | p-value |
|---|---|---|---|
| 29 | 13 | 86% | <0.001 |
| 30 | 15 | 100% | <0.001 |
| 31 | 15 | 100% | <0.001 |
| 32 | 15 | 100% | <0.001 |
| 33 | 15 | 100% | <0.001 |
| 34 | 15 | 100% | <0.001 |
| 35 | 15 | 100% | <0.001 |
| 36 | 15 | 100% | <0.001 |
| 37 | 15 | 100% | <0.001 |
| 38 | 15 | 100% | <0.001 |
| 39 | 15 | 100% | <0.001 |
| 40 | 15 | 100% | <0.001 |
| 41 | 15 | 100% | <0.001 |
| 42 | 15 | 100% | <0.001 |
| 43 | 15 | 100% | <0.001 |
| 44 | 14 | 93% | <0.001 |
| 45 | 15 | 100% | <0.001 |
| 46 | 15 | 100% | <0.001 |
| 47 | 15 | 100% | <0.001 |
| 48 | 15 | 100% | <0.001 |
| 49 | 15 | 100% | <0.001 |
| 50 | 15 | 100% | <0.001 |
| 51 | 15 | 100% | <0.001 |
| 52 | 15 | 100% | <0.001 |
| 53 | 15 | 100% | <0.001 |
| 54 | 15 | 100% | <0.001 |
| 55 | 15 | 100% | <0.001 |
| 56 | 15 | 100% | <0.001 |

TABLE III
THE AMOUNT OF TESTING WEBSITES IN EACH SPOOFING CATEGORY

| Protected | Type of spoofing | Number of websites |
|---|---|---|
| Yes | None | 12 |
| No | Phishing website | 3 |

question written on the main questionnaire form was: "Is this website safe or not (phishing website)". The available options for the testers to select were: Yes, No, and Level of certainty from 1 to 4. where 1 indicated the lowest certainty level The answers written on the main questionnaire and the background questionnaire were recorded for further analysis of utility and usability of our password protection system. The users did not know how many websites are potentially malicious from the set of 15 web pages. Popular websites were cloned using the SiteSucker tool and some were slightly modified to simulate phishing (eg the different types of attacks against our protection system). The tests were performed using their laptops For the follow up user study, the procedure was the same as for the initial user study, however, the participants were not reminded of the instructions for using our protection system. They were asked to visit the same set of websites in a different order and to answer the same question: "Is this website safe or not (phishing website)?".

*2) Results*

Table III shows the number of pages divided into spoofing categories, 20% of all testing websites were trying to mislead the participants by spoofing the UI of other websites. Fig. 4 shows the percentage of websites correctly identified by each tester. Based on the data presented in Fig. 4, the average effectiveness of our extension is 97%, which indicates that the participants were able to assess correctly nearly 97% of the websites they visited (U-l, U-2)t The average level of certainty per website varies from 3.5 to 4, In order to assess the statistical significance of these results. we used a null hypothesis as our point of reference. The null hypothesis says that by randomly guessing the answers, the effectiveness would be equal to 50% since users could either guess "yes" or "no". We assumed a standard threshold.

value (a) of 005. Table II shows the probability (p-value2) of getting either the same or grater effectiveness assuming that the null hypothesis is true, The p-value has been calculated for all participants based on the overall scores they obtained. For 95% of participants, the p-value is less than 0.0011. Table IV presents the p-value

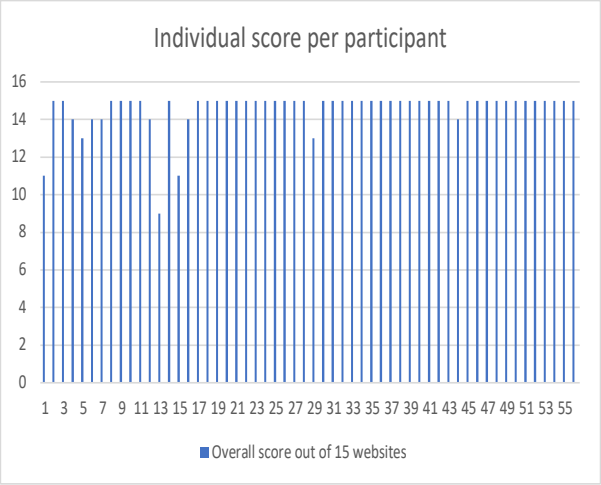


Fig. 4. The overall score of each participant.

calculated per testing website. For all of websites, the p-value does not exceed 0.001.

Table V shows the results of the follow up user study. In the initial user study. these particular participants correctly identified 97% of websites on average. In the follow up user study, the same group of participants correctly identified 16% of spoofing websites on average. We calculated the p-value for each participant using McNeinar's test4. For 95% of participants, the p-value exceeds 0.2; therefore. we (lid not observe a significant decrease in the effectiveness of our protection system. 82% of participants would like to use the our protection system extension in their own browsers. We asked the participants to assess the level of difficulty of using our password protection system. As shown in Table VI, 93% of the participants answered that the browser extension is easy to use while 7% indicated it is rather difficult.

*3) Discussion*

Achieving almost 97% percent efficiency, we have exceeded the percentage indicated in the null hypothesis. which confirms the utility of the extension. We have compared the p-values calculated based on the overall score of each participant to the null hypothesis Only 5 p-Values exceeded the threshold of 0.05. For 92% of participants, the browser extension was useful during the assessment process. Out of 15 websites, the p-mlue for only one website exceeded the threshold of 0.05. This may be caused by the fact that this website was the first phishing website encountered by participants. In summary, even the participants who were not familiar with phishing were able to detect effectively which websites were protected by our protection system in the

TABLE IV
PROBABILITY VALUES CALCULATED FOR WEBSITES' OVERALL SCORES

| Website number | Overall score out of 56 participant | Overall score (%) | p-value |
|---|---|---|---|
| 1 | 56 | 100% | <0.001 |
| 2 | 56 | 100% | <0.001 |
| 3 | 56 | 100% | <0.001 |
| 4 | 54 | 96.4% | <0.001 |
| 5 | 49 | 87.5% | <0.001 |
| 6 | 55 | 98.2% | <0.001 |
| 7 | 55 | 98.2% | <0.001 |
| 8 | 52 | 92.9% | <0.001 |
| 9 | 56 | 100% | <0.001 |
| 10 | 54 | 96.4% | <0.001 |
| 11 | 53 | 94.6% | <0.001 |
| 12 | 55 | 98.2% | <0.001 |
| 13 | 55 | 98.2% | <0.001 |
| 14 | 54 | 96.4% | <0.001 |
| 15 | 56 | 100% | <0.001 |

TABLE V
PARTICIPANTS' OVERALL SCORES FROM THE FOLLOW UP USER STUDY WITHOUT THE EXTENSION

| Participant number | Overall score of spoofing website in initial user study | Overall score of spoofing website in follow up user study | Significant decrease |
|---|---|---|---|
| 1 | 3 | 0 | yes |
| 2 | 3 | 1 | yes |
| 3 | 3 | 1 | yes |
| 35 | 3 | 0 | yes |
| 41 | 3 | 1 | yes |
| 47 | 3 | 1 | yes |
| 49 | 3 | 0 | yes |
| 51 | 3 | 0 | yes |
| 53 | 3 | 1 | yes |
| 55 | 3 | 0 | yes |

TABLE VI
EVALUATION OF THE DIFFICULTY OF USING OUR PROTECTION SYSTEM EXTENSION

| Level of difficulty | Number of Participant | Percentage of Participants |
|---|---|---|
| Very easy to use | 39 | 70% |
| Easy to use | 13 | 23% |
| Difficult to use | 4 | 7% |
| Very difficult to use | 0 | 0% |

presence of active phishing'. Based on the follow up user study without our extension, all participants obtained a significantly worse overall score, which may indicate that our browser extension is highly effective. Surprisingly, some participants mentioned that they do not want to use the browser extension since they are not concerned about the security of their passwords. On the other hand, a number of participants indicated that they would like to know more about the technology before installing this extension.

## V. Conclusion and Future Work

In our proposal we have discussed the weakness of the existing web infrastructure, various vulnerabilities were found out which lead to attacks and hence compromised the security of sensitive information. It is critical to enhance the security of such systems without decreasing performance and usability. The proposed technique mitigates many phishing attacks without introducing much delay or overheads in communication. Other existing solutions require the use of additional servers or they introduce significant performance limitations. For this reason, we decided to develop the mechanisms, which are supported by off-the-shelf hardware. After finishing the user study, we spotted that the extension allows users to easily determine if the website they are visiting is safe or not (Phishing website). The usefulness of our system has been proven by the high scores from the test participants. The vast majority of users were able to differentiate, using our browser extension. This demonstrates that after slight improvements, the system may gain high popularity among users, regardless of their age or profession. This encourages us to continue the research and extend our protection system functionality. Finally, still one significant problem which is not everyone is aware of dangers on the web. According to our research, some people do not even know what phishing is. Therefore, implementing the technical solution may not be enough. Users should take care of their sensitive data and suitable trainings should be provided for them to reassure that.

## References

[1] C. Herley, P. C. Van Oorschot and A. S. Patrick, "Passwords: If We're So Smart, Why Are We Still Using Them?" in *Int. Conf. Financ. Cryptogr. Data Secur.*, in Financial Cryptography and Data Security, in Lecture Note in Computer Science, vol. 5628, pp. 230-237, 2009.

[2] C. Herley and P. Van Oorschot, "A Research Agenda Acknowledging the Persistence of Passwords," in *IEEE Secur. Priv.*, vol. 10, no. 1, pp. 28-36, Jan.-Feb. 2012, doi: 10.1109/MSP.2011.150.

[3] J. Bonneau, C. Herley, P. C. v. Oorschot and F. Stajano, "The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes," *2012 IEEE Symp. Secur. Priv.*, San Francisco, CA, 2012, pp. 553-567, doi: 10.1109/SP.2012.44.

[4] J. Yan, A. Blackwell, R. Anderson and A. Grant, "The memorability and security of passwords – some empirical results," Univ. Cambridge, United Kingdom, Rep. UCAN-CL-TR-500, Sept. 2000.

[5] B. Menkus, "Understanding the use of passwords," *Comput. Secur.*, vol. 7, no. 2, pp. 132-136, Apr. 1988, doi: 10.1016/0167-4048(88)90325-2.

[6] S. Gaw and E. W. Felten, "Password management strategies for online accounts," in *Proc. Second Symp. Usable Priv. Secur.*, July 2006, pp. 44-55, doi: 10.1145/1143120.1143127.

[7] E. F. Gehringer, "Choosing passwords: security and human factors," *IEEE 2002 Int. Symp. Technol. Soci. (ISTAS'02). Soci. Implic. Inf. Commun. Technol. Proc. (Cat. No.02CH37293)*, Raleigh, NC, USA, 2002, pp. 369-373, doi: 10.1109/ISTAS.2002.1013839.

[8] A. K. Jain and B. B. Gupta, "A novel approach to protect against phishing attacks at client side using auto-updated white-list," *EURASIP J. Inf. Secur.*, 2016, no. 9, May 2016, doi: 10.1186/s13635-016-0034-3.

[9] M. He et al, "An efficient phishing webpage detector," *Expert Syst. Appl.*, vol. 38, no. 10, pp. 12018-12027, Sept. 2011, doi: 10.1016/j.eswa.2011.01.046.

[10] Y. Pan and X. Ding, "Anomaly Based Web Phishing Page Detection," *2006 22nd Annu. Comput. Secur. Appl. Conf. (ACSAC'06)*, Miami Beach, FL, 2006, pp. 381-392, doi: 10.1109/ACSAC.2006.13.

[11] R. Islam and J. Abawajy, "A multi-tier phishing detection and filtering approach," *J. Netw. Comput. Appl.*, vol. 36, no. 1, pp. 324-335, Jan. 2013, doi: 10.1016/j.jnca.2012.05.009.

[12] S. C. Jeeva and E. B. Rajsingh, "Intelligent phishing url detection using association rule mining," *Human-Centric Comput. Inf. Sci.*, vol. 6, no. 10, July 2016, doi: 10.1186/s13673-016-0064-3.

[13] Y. Li, R. Xiao, J. Feng and L. Zhao, "A semi-supervised learning approach for detection of phishing webpages," *Optik*, vol. 124, no. 23, pp. 6027-6033, Dec. 2013, doi; 10.1016/j.ijleo.2013.04.078.

[14] S. Roopak and T. Thomas, "A Novel Phishing Page Detection Mechanism Using HTML Source Code Comparison and Cosine Similarity," *2014 Fourth Int. Conf. Adv. Comput. Commun.*, Cochin, 2014, pp. 167-170, doi: 10.1109/ICACC.2014.47.

[15] D. Abraham and N. S. Raj, "Approximate string matching algorithm for phishing detection," *2014 Int. Conf. Adv. Comput. Commun. Inf. (ICACCI)*, New Delhi, 2014, pp. 2285-2290, doi: 10.1109/ICACCI.2014.6968578.

[16] S. Garera, N. Provos, M. Chew and A. D. Rubin, "A framework for detection and measurement of phishing attacks," in *Proc. 2007 ACM Workshop Recurr. Malcode*, Nov. 2007, pp. 1-8, doi: 10.1145/1314389.1314391.

[17] A. Le, A. Markopoulou and M. Faloutsos, "PhishDef: URL names say it all," *2011 Proc. IEEE INFOCOM*, Shanghai, 2011, pp. 191-195, doi: 10.1109/INFCOM.2011.5934995.

[18] E. H. Chang, K. L. Chiew, S. N. Sze and W. K. Tiong, "Phishing Detection via Identification of Website Identity," *2013 Int. Conf. IT Converg. Secur. (ICITCS)*, Macao, 2013, pp. 1-4, doi: 10.1109/ICITCS.2013.6717870.

[19] M. G. Alkhozae and O. A. Batarfi, "Phishing Websites Detection based on Phishing Characteristics in the Webpage Source Code," *Int. J. Inf. Commun. Technol. Res.*, vol. 1, no. 6, pp. 283-291, Oct. 2011.

[20] C. L. Tan, K. L. Chiew and S. N. Sze, "Phishing Webpage Detection Using Weighted URL Tokens for Identity Keywords Retrieval," in *9th Int. Conf. Robot. Vis. Signal Process. Power Appl.*, in Lecture Notes in Electrical Engineering, vol. 398, pp. 133-139, 2017.

[21] R. B. Basnet and T. Doleck, "Towards Developing a Tool to Detect Phishing URLs: A Machine Learning Approach," *2015 IEEE Int. Conf. Comput. Intell. Commun. Technol.*, Ghaziabad, 2015, pp. 220-223, doi: 10.1109/CICT.2015.63.

[22] G. Sonowal and K. S. Kuppusamy, "MASPHID: A Model to Assist Screen Reader Users for Detecting Phishing Sites Using Aural and Visual Similarity Measures," in *Proc. Int. Conf. Inf. Anal.*, Aug. 2016, pp. 1-6, doi: 10.1145/2980258.2980443.

[23] J. Lee, D. Kim and C. Lee, "Heuristic-based Approach for Phishing Site Detection Using URL Features," in *Proc. Third Int. Conf. Adv. Comput. Electron. Electr. Technol.*, Apr. 2015, pp. 131-135.

[24] C. Dhinakaran, D. Nagamalai and J. K. Lee, "Multilayer Approach to Defend Phishing Attacks," *arXiv: 1108.1593*, 2011, [Online] Available: https://arxiv.org/abs/1108.1593