Acta Informatica Pragensia, 2017, 6(2): 114–123
DOI: 10.18267/j.aip.104
**Peer-reviewed paper**

# A Lightweight Anti-Phishing Technique for Mobile Phone

**Abdul Abiodun Orunsolu[*], Misturah Adunni Alaran[*], Adeleke Amos Adebayo[*], Sakiru Oluyemi Kareem[*], Ayobami Oke[*]**

**Abstract**

Mobile phones have become an essential device for accessing the web. This is due to the advantages of portability, lower cost and ease. However, the adoption of mobile phones for online activities is now being challenged by myriads of cybercrimes. One of such crimes is phishing attack. In this work, a lightweight anti-phishing technique is proposed to combat phishing attacks on mobile devices. This is necessary because these mobile platforms have increased the attack surface for phishers while diminishing the effectiveness of existing countermeasures. The proposed approach uses a number of URL behavior to determine the status of a website based on frequency analysis of extracted phishing features from PhishTank. To increase the detection power of unknown pattern, a machine learning algorithm called Support Vector Machine is adopted. The results indicated that the approach is very efficient against phishing sites with negligible false negatives.

**Keywords:** Cyber fraud, Mobile devices, Phishing, Security, URL.

## 1 Introduction

The rapid expansion of e-commerce has witnessed unprecedented adoption by the cyber community with an attendant question on the security of online transactions. This is because a host of online black markets manages by a con artist or hackers now threaten stakeholders' confidence in e-commerce. The incidences of these ugly activities assumed a more dangerous dimension with the advent of phishing, in which both the service providers and unsuspecting users suffer huge financial losses and reputational brand damage (Neupane et al., 2015).

Phishing describes any attempt to deceptively acquire sensitive personal or financial information via electronic communication or fake websites with malicious intent (Hong, 2012). Such fake websites or electronic communication usually comes with the feel or patterns of genuine communication or original sites. Lack of knowledge of security indicators, bounded attention, visual deception and lack of computer system knowledge on the part of unsuspecting users sometimes assists the phishers to have a field day (Alsharnouby et al., 2015). For instance, the Anti-Phishing Working Group (APWG) reported that the total number of unique phishing sites detected from Q1 through Q3 of 2015 was 630,494 (www. apwg.org). This report indicates that phishing is not showing any signs of slowing down in

[*] Department of Computer Science, School of Science and Technology,
Moshood Abiola Polytechnic, Abeokuta, P.M.B 2210 Abeokuta, Ogun State, Nigeria
✉ orunsolu.abdul@mapoly.edu.ng

2015. In the same vein, RSA's security report indicated that phishing attacks cost global organizations $4.5 billion in losses in 2014 (RSA, 2014).

Against the background of this ugly statistics, security expert and research communities have responded with myriads of anti-phishing solutions. These anti-phishing solutions range from users' education to software enhancements (Alsharnouby et al., 2015). While most of these anti-phishing solutions have made a significant difference in dealing with phishing, the proliferation of android/iOS mobile phones in recent times has increased attack surface for phishers (Kumar & Kumar, 2014). This is due to the fact these mobile phones come with some technicalities such as smaller screens, absence of certain bars, customized interfaces etc. which are not considered in most existing anti-phishing solutions. Hence, there is an urgent need to upgrade the anti-phishing countermeasures in the face of advancement of information communication technology. Moreover, the emergence of the ransomware threat, which is usually delivered to a victim's device through phishing messages calls for urgent new anti-phishing countermeasures.

In this paper, we report a lightweight mobile phone-based anti-phishing scheme. The diminished effectiveness of current countermeasures on mobile platforms motivates us to design and implement an android compliant anti-phishing scheme. Our work makes following two research contributions. First, we have proposed a new anti-phishing scheme for mobile phones especially android-based platforms. This approach considers phishing URLs as they represent the identity of a phishing webpage. The proposed approach considers both old and new versions of android OS in order to protect mobile users from phishing URLs. The key feature of this approach is its lightweight as memory utilization is an important consideration on such mobile platforms. Secondly, we used frequency analysis of PhishTank and Alexa dataset to generate our heuristics which are later trained with SVM.

The rest of the paper is organized as follows: Section II discusses literature review and related work. In section III, the proposed architecture is presented and discussed. The experimental results and evaluations are discussed in section IV while section V concludes the work.

## 2 Literature review

This section examines some of the known anti-phishing countermeasures. These countermeasures range from user education or awareness to software enhancements. In user education or awareness approach, end-users are trained to better understand the nature of phishing attacks either by sending regular tips to their email addresses or using a game approach or through specialized training session (Parsons et al. 2015). For instance, Parsons et al. (2015) conducted a role-play experiment of people's ability to differentiate between phishing and genuine emails. In the study, the authors explicitly informed half of the participants about the goal of the study. Their results indicated that informed users were significantly better at discriminating between phishing and genuine emails than the uninformed participants. The main goal of user awareness approach is to reduce the vulnerabilities of a human factor because it does not matter how many defense mechanisms is available if the user behind the keyboard falls for a phish (Hong, 2012).

On the other hand, software enhancements are automated tools developed to bridge the gap that is left due to the human error or ignorance in mitigating phishing attacks. Software enhancement employs a number of attributes such as blacklisting, visual similarity cues, heuristics and multi-channel authentication approach (Khonji, Iraqi, & Jones, 2013).

In blacklist approach, frequently updated lists of previously detected phishing URLs, IP addresses or keywords are maintained to prevent unsuspecting users from accessing those sites. For instance, Prakash et al. (2010) proposed PhishNet to actively predicted new malicious URL from existing blacklist entries. This was achieved by processing URLs and producing multiple variations of the same URL using IP address equivalence, query string substitution, brand name equivalence, directory structure similarity and top level domain replacement.  In the same vein, Google safe browsing is another implementation of blacklist that enables a client application to validate whether a given URL exists in the malicious list that is constantly updated by Google.  Other notable implementations of blacklist approach in browser include VeriSign, Netcraft and Site Advisor. Although blacklist approaches generally have lower false positives, they do not provide protection against zero-hour phishing attacks.

Visual similarity approaches are based on the idea that phishers often design fake pages to have the look and feel of the legitimate sites in order to deceive the unsuspecting users.  To address this difficulty, the visual similarity approach is designed to measure the degree of variation between fake websites and the legitimate sites using visual similarity cues such as text pieces and their style, brand logo and embedded images. Chiew et al. (2015) proposed a method of detecting phishing pages using logo image. The method consists of two processes namely logo extraction and identity verification. In the logo extraction process, the logo image on a site is detected and extracted from all the downloaded image resources of a web page. In the second process, the Google image search is used to retrieve the portrayed identity of the logo image. In this way, the relationship between the query return by Google and the domain name is used to determine the status of the website.

The heuristic approach firstly collects a set of discriminative features (called heuristics) that can separate phishing activities from legitimate ones, then train a machine learning model to predict phishing attacks based on these features and finally use the model to recognize phishing attack in the real world (Pan & Ding, 2006). In heuristics approaches, web pages or emails are treated as documents in which features can be extracted from the header, content, and body. Gowtham & Krishnamurthi, 2014 presented a heuristic based approach using a service-oriented three-layer architecture. The service-oriented three layers consist of client interface layer, web service middleware layer and anti-phishing component layer. The anti-phishing component layer provides a set of reusable components to convert a webpage into feature vectors using finest heuristic methods and external repositories of information. The feature vectors are fed into trained support vector machine classifier to generate phishing label.

Multi-channel authentication approach is focused on the prime motivation of phishers which is to obtain sensitive credentials of online users (Purkait, 2012). With such information leaked to phishers, online users can suffer damages ranging from financial losses to unhealthy disclosure of their secrets. Therefore, it is imperative to avoid theft of credentials with a seamless authentication protocol in which genuine users can be protected from phishers. One of such method is the use of One-Time –Password (OTP). Huang et al. (2011) investigated the use of OTP in mitigating phishing attack. In their study, an OTP is delivered on demand to parties through a reliable secondary communication channel. On the receipt of the OTP, the user can log in before the password expired. The approach involves two processes namely a registration process and a login process. In the registration process, a user chooses a unique account name, select a login password, fill in all the required information fields, complete and additional instant message account registration and provide at least one type of personal contact information. In the login process, the registered user can log in with the OTP assigned by the website.

## 2.1 Smartphones, URL, Phishing and Ransomware

According to the European Union Agency for Network and Information Security (ENISA) report 2016, there are a number of reasons why the risk of phishing is important for smartphone users. First, smartphones have a smaller screen where attackers can easily disguise trust cues that users rely on to decide before submitting credentials. Second, app stores provide a new way of phishing by allowing attackers to place fake apps in the app store. Third, smartphone provide additional channels that can be used for phishing e.g. SMS. Lastly, smartphones are a new type of device and users may not be aware of the fact that phishing is a risk on smartphones as well. Sadly, the risk presented in this report is further aggravated with decommissioned smartphones in which large amounts of sensitive information is left unformatted on recycled phones.

Mobile users' habits and preferences further ease the mobile phishing attacks. During the past few years, touchscreen smartphones have become dominant in the mobile phone market. However, typing on a virtual keyboard is not as easy as on a physical keyboard due to lower input accuracy, particularly when walking or sitting in a moving vehicle. Because of that, it is tempting to follow links in web pages or e-mails rather than typing the links manually. Another factor is that on smartphones, switching among applications or even shifting to other pages within a browser, is more complicated and tedious than when performed on a PC. Users who value convenience usually prefer to follow links from other applications. In addition, phishing attacks can succeed because users become accustomed to entering their credentials in familiar, repeated login interfaces. If users frequently encounter legitimate links whose targets prompt them for private data, then users get used to reflexively supplying the requested data.

Recently, phishers develop ransomware which executes a crypto virology attack that adversely affects computing devices and prevent access until the victim pays a ransom. Today, the threat of ransomware is a global problem and 93% of phishing emails are ransomware-based (CSO, 2016; Richardson & North, 2017). Most ransomware propagation techniques bear similarity with phishing proliferation techniques. Basically, both ransomware and phishing use traffic redirection, email attachments, social engineering, popup advertisement et cetera (Bhardwaj et al., 2017). Therefore, the design of an anti-phishing countermeasure is a step forward in mitigating the ravaging effect of ransomware.

Against this background, it becomes imperative to investigate a lightweight anti-phishing scheme for smartphone users. This approach is based on heuristics especially the URL behaviour. This is because spam SMS has transformed to phishing SMS where a genuine content contains a malicious link. This message or SMS may be without any body of message except the malicious URL in it, thereby making it difficult for content-based filter detection. This URL leads to the actual phishing page which is clones of legitimate websites and lure the users into entering sensitive information. The attackers pose various critical conditions such as account suspension, failed transaction, urgent relief donations for natural disaster victims etc. Sometimes, phishers can disguise their attacks as wedding invitations or tax rebates to lure their victims.

# 3 Proposed scheme

The proposed scheme of a lightweight smartphone anti-phishing scheme based on URL behavior. Phishing URLs can be analyzed based on the lexical features and host-based features. Knowing that phishers use visual components ripped off from a legitimate web page in their website to deceive unsuspecting users, this motivated us to propose an anti-phishing scheme based on URL. No matter how a phisher perfectly clones a website, the URL cannot be cloned exactly the same way.

The URL behaviors use in this study are formulated based on detailed analysis of 1000 confirmed phishing sites from PhishTank and 500 top clean sites from Alexa. PhishTank is a community-based database with a reliable phishing dataset. PhishTank provides data for download or access via an API call under a restrictive license. On the other hand, Alexa provides commercial web traffic data, global rankings, and analytics of about 30 million websites (Alsharnouby et al., 2015).

After extensive analysis of phishing URLs and genuine URLs, we formulate certain categories of heuristics under which classification can be made. A unique filter is associated with each heuristic and the detection rules are formulated using the various combinations of these filters. In addition, each filter is associated with a certain value using Support Vector Machine learning system. To represent each URL, a vector with a true/false value is created after the identification of all the lexical in the URL. Table 1 presents the frequency of the various heuristics extracted from 1000 phishing URLs from PhishTank.

The basic architecture of the proposed lightweight anti-phishing scheme is presented in Figure 1. The workflow of the architecture consists of a module to fetch URL, a module to filter URLs based on behavioral features captured in the frequency analysis and a trained SVM classifier which triggered an android detector module.

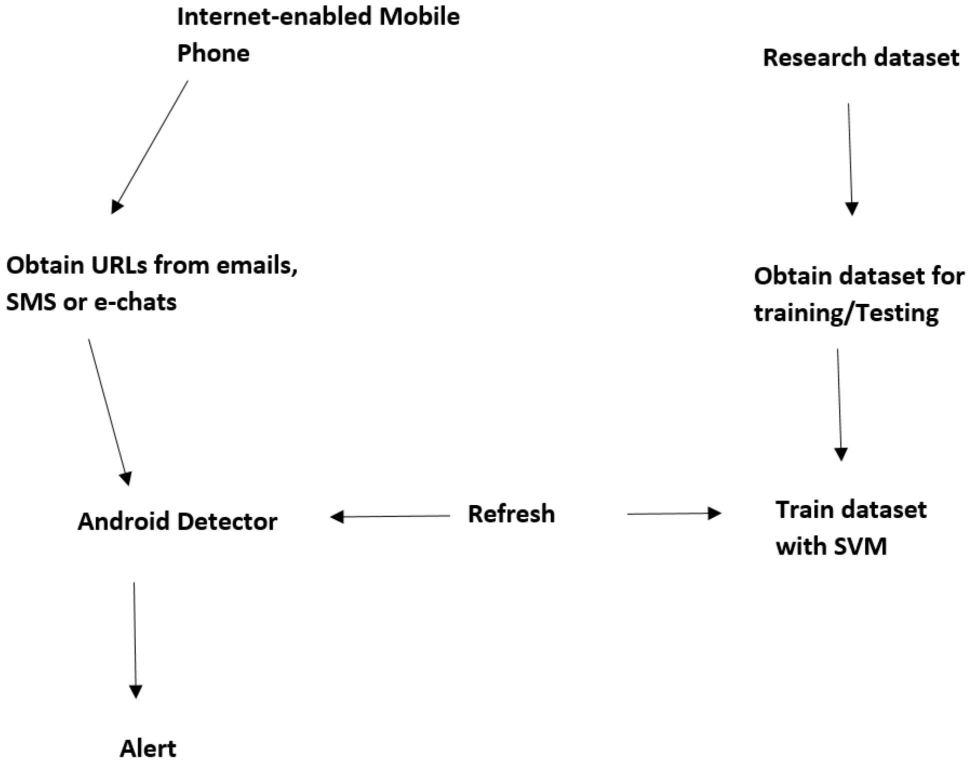

Internet-enabled Mobile Phone
Research dataset
Obtain URLs from emails, SMS or e-chats
Obtain dataset for training/Testing
Android Detector
Refresh
Train dataset with SVM
Alert

***Fig. 1**. Proposed Lightweight Mobile Anti-Phishing Scheme. Source: Authors.*

In the fetch URL module, the system first fetches new URLs via email (a), SMS (b) or social networking sites (c). The algorithm 1 is designed to intercept a new URL before the user visits the web page through activation of a required package. The following pseudo code illustrates the tasks of the fetch module:

| S/N | Phishing Features | No of Appearance per 1000 |
|---|---|---|
| 1 | URL with  more than three “/” | 315 |
| 2 | More than three “.” | 108 |
| 3 | Length > 30 | 585 |
| 4 | Using “@” | 9 |
| 5 | Using “-” | 252 |
| 6 | Using IP | 10 |

***Tab. 1.** High Impact features on URL phishing instances. Source: Authors.*

```
Load the apkFileName
Set isFrameworkApk: False
Use permission android:name =”android.permission. INTERNET”,
“android.permission.ACCESS_WIFI_STATE”, “android.permission.
ACCESS_NETWORK_STATE”
GetURL (a, b, c )
If android:configChanges =”email, SMS,MySocialNetworkList” is found
```

***Alg. 1.** Algorithm for fetching URL on Mobile device. Source: Authors.*

In addition, two URL features are included without estimating their instances of occurrence. These are:

a. **Abnormal URL shortening**: Phishers use URL shorteners to obfuscate phishing URLs especially on social networking sites. If the link shortening timestamp patters and the Number of encoders are not similar to genuine Bitly URL, then it is likely to be phishing. Otherwise, it is legitimate
b. **Malicious file download extension**: Most phishing sites or email contain an instruction to download certain files which are ransomware-based. If the file contains specified extension such as .aaa, .abc or 6-7 length extension of random characters, then it is phishing and suspicious. Otherwise, it is legitimate

These features are included to increase the accuracy of detection approach to tackle the current problems of malicious URL shortener and ransomware.

In the next module, the application feature set is compared for any mismatch in the retrieved URL (Algorithm 2). This feature set is obtained from the frequency analysis of PhishTank and Alexa. It includes the length of the hostname, the length of the URL, number of dots, presence of suspicious characters and other special binary characters. This analysis produces a pattern which is arranged into rules. After analyzing the dataset of 1000 phishing sites and 500 top clean sites from Alexa, the percentage of web page matching the lexical features are listed in Table 2.

| Feature | Legitimate | Phishing |
|---|---|---|
| Using IP Address | 0% | 99% |
| Length>30 | 0% | 59% |
| Using more than three “/” | 0% | 68% |
| More than 3 “.” | 0.01% | 89% |
| Using “@” | 0% | 99% |
| Using “-” | 0.01% | 75% |

***Tab. 2.*** *Analysis of Lexical features matching with phishing sites. Source: Authors.*

The following pseudo code illustrates the task of this module.  The code retrieves all the available links found in the getURL module and runs a check link feature function to scan for any possible characteristic or behavior of the link. After obtaining the behavior of the URL, the algorithm checks for any misrepresentation in the link from feature analysis of the research database.

In the final step, the system invokes a trained SVM classifier. In order to classify the different features of URL, we apply a method that analyzes data and recognizes pattern using LibSVM library. The LibSVM uses a dimension feature vector *Vp* deduced from the feature generation steps, representing URL structural and behavioral patterns. Each feature that SVM learns from needs a value or weight using represented as true i.e. phishing (1) or false i.e. legitimate (0). The classification of URL in SVM is a binary classification problem with only two possible cases i.e. legitimate or phishing.  A typical treat train model of SVM contains the following parameter

```
Check for URL-filter match
Retrieved the fetched URL From GetURL (a, b, c)
Check LinkFeature ()
d: Set Phishtank [URL_filter]^Alexa[URL_filter]
Compare with d
var scan = document.getElementById( 'txtScan ');
var filter = /^([a-zA-Z0-9_\.\-])+
\@(([a-zA-Z0-9\-])+\.)+([a-zA-Z0-9]{2,4})+$/
;if (!filter.test( scan.value)) {
alert( 'This contains illegal characters' );
if (!filter.test( scan.value)) {
alert ( 'This contains illegal characters' );
```

***Alg. 2.*** *Algorithm for comparing URL. Source: Authors.*

To ensure the currency of the trained data, a refresh function is used between the train data and test data in the android detector. The android detector is the application logic which checks URL status with the SVM classifier and triggered appropriate message or alert.

```
svm_type c_svc
        kernel_type rbf
           gamma 0.0454545
            nr_class 1
               total_sv 0
                      rho
               label 1
                  nr_sv 0
                SV
```

***Alg. 3.*** *SVM algorithm for class labelling. Source: Authors.*

# 4 Implementation and evaluation

We have implemented the lightweight anti-phishing URL verifier for mobile phone on Android OS. We run the system on different hardware platforms whose specifications are provided in Table 3. The speed of the system is the time taken for each smartphone configuration to detect URL retrieved by the system. The average response time for all the five devices is 2s.

| Smartphone (SM) Type | Processor | RAM | OS | Screen size | Speed of the system Detection |
|---|---|---|---|---|---|
| Gionee SM | 800Mhz | 512MB | Android 4.0 | 4.0 Inches | 4s |
| Techno L5 | 1.3 Ghz | 1 GB | 5.1.1 lollipop | 4.5 Inches | 1s |
| Toshiba AT7-C | 1.5 Ghz | 1 GB | Android 4.4.2 | 7 Inches | 1s |
| Nokia X | 1 Ghz | 512 MB | Nokia 4.1.2 | 4 Inches | 2s |
| Techno H5 | 1 Ghz | 512 MB | Android 4.2.2 | 4 Inches | 2s |

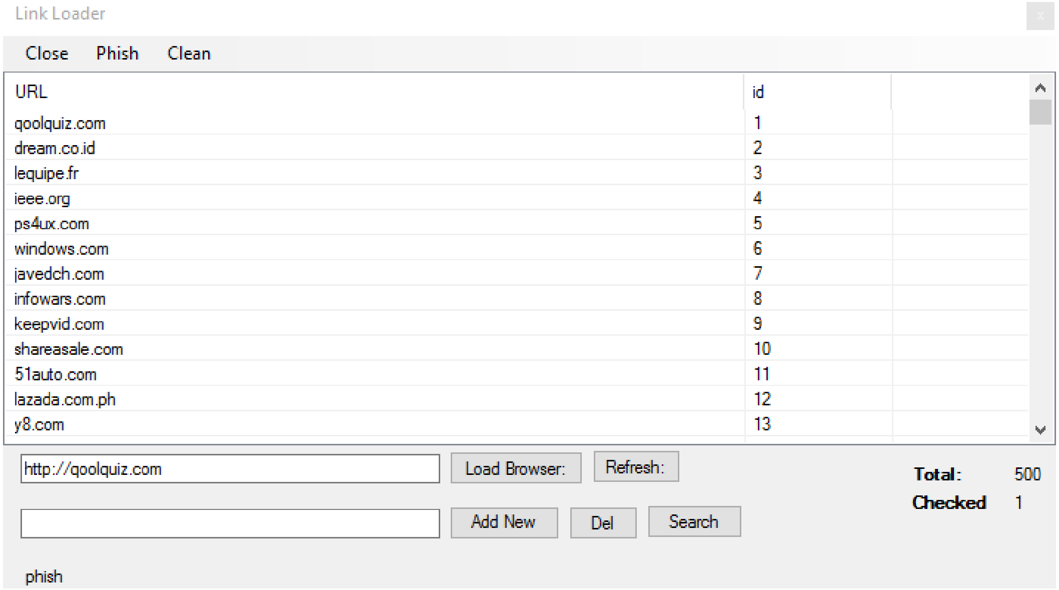
***Tab. 3.*** *Summary of the system performance. Source: Authors.*

Figure 2 is a simple graphical user interface of the system showing the detection of a phishing site. We tested the system on 3530 URLs consisting of a dataset from PhishTank, Alexa, and randomly selected websites through live streaming from the Internet (Table 4). The datasets from PhishTank and Alexa are automatically fed into the browser using a Link Loader program to speed up the evaluation process (Figure 2).

| S/N | Sources of testing data/Numbers of URL | Detection Rate |
|---|---|---|
| 1 | PhishTank/2200 | 2126/2200 = 96.6% |
| 2 | Alexa/1000 | 989/1000 = 98.9% |
| 3 | Randomly live stream from Internet/330 | 326/330 = 98.8% |

***Tab. 4.*** *Summary of Testing Dataset and Detection Rate. Source: Authors.*

The high detection rate of the randomly selected live URL stream from the Internet may be connected with the fact that all the URLs in this category are English-based. However, the slight decrease in the detection rate of the PhishTank dataset were due to the fact that some of the non-English based websites were wrongly classified as legitimate.



| URL | id |
|---|---|
| qoolquiz.com | 1 |
| dream.co.id | 2 |
| lequipe.fr | 3 |
| ieee.org | 4 |
| ps4ux.com | 5 |
| windows.com | 6 |
| javedch.com | 7 |
| infowars.com | 8 |
| keepvid.com | 9 |
| shareasale.com | 10 |
| 51auto.com | 11 |
| lazada.com.ph | 12 |
| y8.com | 13 |

http://qoolquiz.com

Total: 500
Checked 1

phish

***Fig. 2.*** *Interface of the Link Loader Program. Source: Authors.*

## 5 Conclusion and future works

There has been an aggregative rise in phishing attacks in the past couple of years; however, there seems to be no solution to subvert such threats. Recently there is a rise in the number of internet users, and more people are joining the trend every day. With this, companies are beginning to have an online presence, which makes the user or customers expose sensitive information such as their username and password, financial accounts. This anti-phishing scheme is designed and implemented for mobile users.

The advantage of this anti-phishing is to protect web users against a spoofed website. The new system is reliable and efficient. In future, this proposed approach can be improved by setting more powerful rules to detect more URL and to add website source code scanner to detect a phishing page not only with the URL but also the source code of the website. In addition, we hope to remove the problem of language dependence of the current approach to handle non-English based URLs and webpages.

## References

**Neupane, A., Rahman, M.L., Saxena, N., & Hirshfield, L.** (2015). A Multi-Modal Neuro-Physiological Study of Phishing Detection and Malware Warnings. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security* (pp. 479-491). New York: ACM. doi: 10.1145/2810103.2813660

**Bhardwaj, A., Subrahmanyam, G.V.B., Avasthi, V., & Sastry, H.** (2016). Ransomware: A Rising Threat of new age Digital Extortion. Retrieved from https://arxiv.org/abs/1512.01980

**Chiew, K. L., Chang, E. H., Sze, S. N., & Tiong, W. K.** (2015). Utilization of website logo for phishing detection. *Computers and Security*, 54(October), 16-26. doi: 10.1016/j.cose.2015.07.006

**CSO.** (2016). *CSO: Online report on phishing activities*. Retrieved from http://www.csoonline.com/articles

**Gowtham, R., & Krishnamurthi, I.** (2014). A Comprehensive and efficacious architecture for detecting phishing pages. *Computers and Security*, 40(February), 23-37. doi: 10.1016/j.cose.2013.10.004

**Hong, J.** (2012). The State of phishing attacks. *Communications of the ACM*, 55(1), 74-81. doi: 10.1145/2063176.2063197

**Huang, C., Ma, S., & Chen, K.** (2011). Using one-time passwords to prevent password phishing attacks. *Journal of Network and Computer Applications*, 34(4), 1292-1301. doi: 10.1016/j.jnca.2011.02.004

**Parsons, K., McCormac, A., Pattinson, M., Butavicius, M., & Jerram, C.** (2015). The design of phishing studies: Challenges for researchers. *Computers & Security*, 52(July), 194-206. doi: 10.1016/j.cose.2015.02.008

**Khonji, M., Iraqi, Y., & Jones, A.** (2013). Phishing Detection: A Literature Survey. *IEEE Communications Surveys & Tutorials*, 15(4), 2091-2121. doi: 10.1109/SURV.2013.032213.00009

**Kumar, G., & Kumar, K.** (2014). Network Security – an updated perspective. *Systems Science & Control Engineering*, 2(1), 325-334. doi: 10.1080/21642583.2014.895969

**Alsharnouby, M., Alaca, F., & Chiasson, S.** (2015). Why phishing still works: User strategies for combating phishing attacks. *International Journal of Human-Computer Studies*, 82(October), 69-82. doi: 10.1016/j.ijhcs.2015.05.005

**Pan, Y., & Ding, X.** (2006). Anomaly based web phishing page detection. In *Proceedings of the 22nd Annual Computer Security Applications Conference*. New York: IEEE. doi: 10.1109/ACSAC.2006.13

**Prakash, P., Kumar, M., Kompella, R., & Gupta, M.** (2010). PhishNet: Predictive blacklisting to detect phishing attacks. In *Proceedings IEEE of the INFOCOM, 2010*. New York: IEEE. doi: 10.1109/INFCOM.2010.5462216

**Purkait, S.** (2012). Phishing counter measures and their effectiveness – literature review. *Information Management & Computer Security*, 20(5), 382-420. doi: 10.1108/09685221211286548

**Richardson, R., & North, M.** (2017). Ransomware: Evolution, Mitigation and Prevention. *International Management Review*, 13(1), 10-21.

**RSA.** (2014). RSA monthly online fraud report. *Anti-Fraud Command Center*. Retrieved from https://www.rsa.com/content/dam/rsa/PDF/rsa-online-fraud-report-0914.pdf