International Journal of Engineering & Technology, 7 (4.31) (2018) 1-6

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Identifying the Most Effective Feature Category in Machine Learning-based Phishing Website Detection

Choon Lin Tan, Kang Leng Chiew[1]*, Nadianatra Musa[2], Dayang Hanani Abang Ibrahim[3]

Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak, 94300 Kota Samarahan, Sarawak, Malaysia
**Corresponding author E-mail: klchiew@unimas.my*

Abstract

This paper proposes an improved approach to categorise phishing features into precise categories. Existing features are surveyed from the current phishing detection works and grouped according to the improved categorisation approach. The performances of various feature sets are evaluated using the C4.5 classifier, whereby the content URL obfuscation category is found to perform the best, achieving an accuracy of 95.97%. Additional benchmarking is conducted to compare the performance of the winning feature set against other feature sets utilised in existing phishing detection techniques. Results suggest that the winning feature set is indeed an effective feature category which has contributed significantly to the performance of existing machine learning-based phishing detection systems.

Keywords: *Classification; Feature Categorisation; Machine Learning; Phishing Detection; Web Security*

1. Introduction

Phishing is a cyber-threat that utilises counterfeit websites to steal sensitive user information such as account login credentials, credit card numbers, etc. Victims are usually led to the phishing websites by clicking on a URL in a fraudulent email that claims to originate from reputable institutions. At the phishing website, victims will then be presented with familiar visual cues (e.g., logo, colour, design, etc.) to convince them to submit their personal information.

Over the years, the severity of phishing attacks has not seen any significant decline despite mitigation efforts such as increasing public's awareness and deploying technical security solutions. As of June 2017, the Anti-Phishing Working Group (APWG) has reported that a number of unique phishing websites remained high at 50,720 [1]. In another report by RSA, it is estimated that global organisations suffered $9 billion loss due to phishing incidents in 2016 [2]. As a result, users are hesitant to fully utilise online banking and e-commerce services.

The blacklist-based detection system is among the most widely deployed anti-phishing solutions in conventional browsers such as Google Chrome and Mozilla Firefox. The blacklist-based detection system queries a central database of known phishing URLs and issues a warning when the user navigates to a known phishing website. However, recent studies have shown that blacklist-based solutions are unable to capture newly launched phishing websites [3], [4].

Another established form of the anti-phishing solution is the machine learning-based detection system. This technique is considered as state-of-the-art due to its' ability to recognise even new phishing websites. Machine learning-based detection systems rely on classifiers which function as decision systems to detect phishing based on features harvested from a variety of sources such as webpage URL, HTML contents, third party services, etc.

The selection of effective features is crucial in developing high-performance machine learning-based phishing detection systems. Many existing researchers adopt features that appeared commonly in prior phishing detection studies. They tend to consider the existing common features as good features, even without established experimental results to support such belief. On the other hand, some researchers may propose additional or new features to enhance their phishing detection system [5]–[9].

In addition, the performance of features is rarely assessed by category-based benchmarking. As a result, anti-phishing researchers may labour in vain when focusing on certain feature categories that are less effective, thus failing to attain the desired phishing detection performance. Hence, establishing proper category-based benchmarking is important so that security experts can concentrate their efforts on a superior feature category that has more potential to improve the phishing detection rate. Moreover, capitalising on superior feature category facilitates rapid development of efficient yet effective anti-phishing applications.

Thus, in this paper, we surveyed the features employed in existing phishing detection works and proposed an improved categorisation approach to classify them. Through the experiments, we provide benchmarking results to compare the performance of different feature categories. Additionally, we conducted distribution analysis on the features' values to provide more insight as to why certain feature categories are better in differentiating between phishing and legitimate websites. In summary, the main contributions of this paper are highlighted as follows:

a) Proposing a new categorisation approach to group phishing features into more precise categories.
b) Conducting benchmarking to assess and compare the performance of each feature category.
c) Identifying a small set of superior features from the winning category that achieves a high phishing detection rate while effectively minimising computational processing power.

The remainder of this paper is organised as follows: Section 2 introduces related studies on feature evaluation from existing machine learning-based phishing detection techniques. Section 3 lists the improved phishing feature categories. Section 4 describes the experimental setup, the results and findings. Finally, Section 5

highlights the contribution of our work and concludes this paper.

2. Related Works

In this section, we provide an overview of the existing anti-phishing studies related to feature evaluation. One of the earliest works in feature analysis for phishing detection was established by Garera et al. [10]. A preliminary study was carried out to analyse various URLs used in phishing attacks, which motivated the authors to propose a set of 18 URL features. These features were categorised into four groups, namely reputation-based, whitelist-based, obfuscation-based and sensitive word-based categories. The experiment was conducted using a logistic regression classifier, which achieved an accuracy of 97.31%. Though the results seemed promising, this method can be easily circumvented by intentionally manipulating the phishing URLs. In addition, most of the features were obtained from specific Google data sources, which are susceptible to rapid changes and possible deprecation that may render the proposed phishing detection system inoperable.

In another phishing detection work, Xiang et al. [11] proposed CANTINA+, which consists of 15 features from various categories such as URL, HTML content, search engine and third-party services. The authors combined their newly proposed features with several existing features taken from [10] and [12]. A Bayesian Network classifier is employed to evaluate the performance of the proposed method, achieving 92% true positives and 0.4% false positives. Moreover, the features were assessed individually by using the Area Under the Curve (AUC) of the Receiver Operating Characteristic (ROC) curve. Results suggest that features derived from HTML content performed the best, followed by the search engine and third-party services. Their feature analysis revealed that a number of URL features from [10] were no longer effective in tackling phishing strategies which could have evolved over the years.

In [13], Abdelhamid et al. proposed a total of 16 features to be used in their novel classification algorithm called Multi-label Classifier based Associative Classification (MCAC) to detect phishing websites. These features were derived from webpage URL, HTML content, and third-party services. An initial feature assessment is conducted to validate the usefulness of each selected feature. In particular, the authors ran a frequency analysis experiment that counts each feature over the total samples in the collected dataset. Their analysis results seem to agree with a similar study by Mohammad et al. [14], which suggest that features from certain categories may have greater capabilities in distinguishing between phishing and legitimate websites. For example, in [14], the address bar feature category was found to be more influential. In our work, we employ a different analysis technique to gain better insights as to why certain feature categories are more effective, where the results are shown in Section 4.

Ramesh and Krishnamurthi [15] proposed a set of website identity features and combined it with existing features commonly used in the literature, resulting in a total of 15 features. These features were categorised into six groups, whereby parallel processing was leveraged to extract features in each group simultaneously. An SVM classifier is then used to train and classify the features, achieving true positives and false positives of 98.24% and 1.71% respectively. Additionally, individual feature analysis using AUC of ROC curve has shown that features derived from website identity and third-party services are among the most influential ones.

Recently, Moghimi and Varjani [9] adopted a set of nine existing features derived from webpage URL and introduced another 8 features related to URLs in the webpage content. The newly proposed features focus on quantifying the string similarity and SSL state of the webpage resource elements. An SVM classifier is employed as the classification model, which achieved 99.14% true positives and 0.86% false negatives. To further assess the impact of each proposed feature towards classification performance, features were excluded one at a time, and the reduction in detection accuracy was observed. However, it is less effective to assess features through single elimination technique, as other powerful features may still exist in the classification model which helps to maintain the accuracy.

Zuhair et al. [16] explored 48 new features focusing on specific source codes in the HTML content, and another 10 features that were based on common URL features from existing works. The 2 features categories were evaluated using an SVM classifier. Results suggest that the performance of URL features and HTML content features are comparable, differing both in true positives and true negatives by less than 1%. Furthermore, features were also assessed by combining both categories that achieved better performance compared to the individual categories. However, it is more intuitive to benchmark features by grouping them into finer categories, which is one of the contributions in this paper.

3. Methodology

In this section, the feature preparation method and the improved categorisation approach is presented.

3.1. Features Preparation

Features employed in existing phishing website detection studies can be sourced internally or externally. Internal features can be directly extracted from the webpage URL and HTML source codes, while external features are obtained from querying third party services such as domain registry, search engine, WHOIS records, etc. In this work, we focus solely on internal features for the following reasons:

a) The webpage URL and HTML source codes are more likely to be available in most phishing datasets. Using commonly accessible features ensures that our study is relevant to most anti-phishing researchers.
b) Third party services are unstable and unpredictable. For example, the public API for querying Google PageRank metric was officially deprecated in 2016 [17], thus affecting many phishing detection techniques that depend on the PageRank feature [10], [11], [15], [18].

Overall, a total of 43 features were extracted from webpages URLs and HTML sources codes. These features were selected after a comprehensive review of related works on machine learning-based phishing website detection. It is based on 18 research papers published between 2006 and 2016. The full list of features is provided in Section 3.2.

3.2. Features Categorisation

This subsection describes our improved approach to categorise 43 features extracted in the previous phase. In the following subsections, we elaborate on the definitions and characteristics that make up each category and introduce the related features.

3.2.1. Category 1 - Content URL Obfuscation

This category is intended to capture phishing patterns which obfuscate URLs in HTML content such as hyperlinks, resource links, etc. These links have unique patterns of usage in phishing websites that differentiate them from legitimate websites. In particular, the HTML content of phishing websites tends to include a high number of external URLs to mislead users or to pull in resources (e.g., images, CSS file, etc.) from the original website. Table 1 lists the features related to content URL obfuscation.

Table 1: Features related to content URL obfuscation

Identifier	Description
PercentExternalHyperlinks	Counts the percentage of external hyperlinks in webpage HTML source code [6], [14], [19]–[21].
PercentExternalResourceUrls	Counts the percentage of external resource URLs in webpage HTML source code [6], [14], [19].
ExternalFavicon	Checks if the favicon is loaded from a domain name that is different from the webpage URL domain name [19].
ExternalFormAction	Checks if the form action attribute contains a URL from an external domain [14], [19].
PercentNullSelfRedirectHyperlinks	Counts the percentage of hyperlinks fields containing an empty value, a self-redirect value such as "#", the URL of a current webpage, or some abnormal value such as "file://E:/" [6], [11], [19], [22].
FakeLinkInStatusBar	Checks if HTML source code contains JavaScript command onMouseOver to display a fake URL in the status bar [14], [19].
AbnormalExternalFormAction	Check if the form action attribute contains a foreign domain, "about blank" or an empty string. Apply rules to generate value [19].

3.2.2. Category 2 - Domain Obfuscation

Domain obfuscation category targets phishing patterns that obfuscate the domain name segment of the webpage URL. We also check other segments of the webpage URL for possible obfuscation with domain related tokens. The phisher's main purpose of obfuscating the aforementioned URL segments is to deceive novice users with words or token that resembles legitimate URLs. The features related to domain obfuscation are listed in Table 2.

Table 2: Features related to domain obfuscation

Identifier	Description
NumDashInHostname	Counts the number of "-" in hostname part of webpage URL [11], [14], [19], [23].
IpAddress	Checks if IP address is used in hostname part of webpage URL [6], [10], [11], [14], [19], [20], [24].
DomainInSubdomains	Checks if TLD or ccTLD is used as part of the subdomain in webpage URL [11].
DomainInPaths	Checks if TLD or ccTLD is used in the path of webpage URL [6], [11], [19].
HttpsInHostname	Checks if HTTPS is obfuscated in hostname part of webpage URL.
EmbeddedBrandName	Checks if the brand name appears in subdomains and path of webpage URL [11]. Brand name here is assumed as the most frequent domain name in the webpage HTML content.
FrequentDomainNameMismatch	Checks if the most frequent domain name in HTML source code does not match the webpage URL domain name.

3.2.3. Category 3 - HTML Content

This category targets miscellaneous phishing characteristics that may occur in the HTML content. The features related to HTML content are listed in Table 3.

Table 3: Features related to HTML content

Identifier	Description
InsecureForms	Checks if the form action attribute contains a URL without HTTPS protocol [11].
RelativeFormAction	Checks if the form action attribute contains a relative URL [11].
AbnormalFormAction	Check if the form action attribute contains a "#", "about:blank", an empty string, or "javascript:true" [19].
RightClickDisabled	Checks if HTML source code contains JavaScript command to disable right click function [14], [19].
PopUpWindow	Checks if HTML source code contains JavaScript command to launch pop-ups [16], [25].
SubmitInfoToEmail	Check if HTML source code contains the HTML "mailto" function [19].
IframeOrFrame	Checks if iframe or frame is used in HTML source code [19].
MissingTitle	Checks if the title tag is empty in HTML source code [7].
ImagesOnlyInForm	Checks if the form scope in HTML source code contains no text at all but images only.

3.2.4. Category 4 - Symbol Exploit

This category of features checks the webpage URL for the abuse of symbols, characters, and sensitive words. The features related to symbol exploit are listed in Table 4.

Table 4: Features related to symbol exploit

Identifier	Description
NumDots	Counts the number of dots in webpage URL [13], [26], [27].
NumDash	Counts the number of "-" in webpage URL [11], [14], [19], [23].
AtSymbol	Checks if "@" symbol exist in webpage URL [6], [11], [14], [19], [23].
TildeSymbol	Checks if "~" symbol exists in webpage URL [7].
NumUnderscore	Counts the number of "_" in webpage URL [23].
NumPercent	Counts the number of "%" in webpage URL [7].
NumAmpersand	Counts the number of "&" in webpage URL [7].
NumHash	Counts the number of "#" in webpage URL [7].
NumNumericChars	Counts the number of numeric characters in webpage URL [23].
DoubleSlashInPath	Checks if "//" exist in the path of webpage URL [15].
NumSensitiveWords	Counts the number of sensitive words (i.e., "secure", "account", "webscr", "login", "ebayisapi", "sign in", "banking", "confirm") in webpage URL [10], [11].

3.2.5. Category 5 - Webpage URL Properties

This category captures the structural characteristics of the webpage URL such as the number of characters or tokens in specific URL segments. Phishers may design a URL with unusual structural properties as a visual deception tactic. The features related to webpage URL properties are listed in Table 5.

Table 5: Features related to webpage URL properties

Identifier	Description
SubdomainLevel	Counts the level of the subdomain in webpage URL [6], [11].
PathLevel	Counts the depth of the path in webpage URL [23].
UrlLength	Counts the total characters in the webpage URL [14], [19].
NumQueryComponents	Counts the number of query parts in webpage URL [7].
NoHttps	Checks if HTTPS exist in webpage URL [6], [9], [14], [19], [23].
RandomString	Checks if random strings exist in webpage URL [7].
HostnameLength	Counts the total characters in hostname part of

	webpage URL [9].
PathLength	Counts the total characters in the path of webpage URL [9].
QueryLength	Counts the total characters in the query part of webpage URL [9].

4. Results Analysis and Discussions

4.1. Dataset Preparation

To gather the required features, we collected webpages from January to May 2015 and from May to June 2017. Specifically, 5000 phishing webpages were downloaded based on URLs from PhishTank [28] and OpenPhish [29], furthermore, another 5000 legitimate webpages were downloaded based on URLs from Alexa [30] and the Common Crawl [31] archive. The complete HTML documents were saved to local storage together with the related resources (e.g., images, CSS files, JavaScript files) to enable offline rendering in the browser. The downloaded webpages were manually filtered to remove incomplete, defective or duplicate instances of the webpages. To automate our feature extraction, we employed the Selenium WebDriver [32], a script-based automation framework for the browser. A series of short scripts written in Python programming language is executed, which in turn will direct the browser to load the webpage, extract the feature value, and save it to a text file.

4.2. Experimental Setup

The categorised features were trained and tested using the Waikato Environment for Knowledge Analysis (Weka) [33] machine learning software. We have deliberately used only a single classifier, namely C4.5 for evaluation of all feature categories to ensure a consistent classification performance. In the training phase, 70% of the dataset is utilised while the remaining 30% is reserved for testing phase. The classification accuracy, ACC is used as the evaluation metric, and is computed as follows:

$$\mathrm{ACC} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}, \quad (1)$$

whereby TP, TN, FP, and FN denote true positive, true negative, false positive and false negative respectively. Experiments were conducted on a desktop computer equipped with an Intel Core i5 2.5 GHz CPU, 6 GB RAM and Windows 7 Home Premium 64-bit operating system.

4.3. Evaluation I — Performance Comparison between Different Feature Categories

This evaluation benchmarks the performance of five different feature categories using the C4.5 classifier. The outcome of this evaluation will help anti-phishing researchers to capitalise on a superior feature category to further improve detection accuracy. Results are shown in Table 6.

Table 6: Performance of different feature categories on C4.5 classifier

Feature Category	TP (%)	FN (%)	TN (%)	FP (%)	ACC (%)
Content URL obfuscation	95.27	4.73	96.67	3.33	95.97
Domain obfuscation	92.80	7.20	49.93	50.07	71.37
HTML content	58.00	42.00	94.33	5.67	76.17
Symbol exploit	77.07	22.93	87.00	13.00	82.03
Webpage URL properties	69.07	30.93	83.20	16.80	76.13

Results suggest that the content URL obfuscation category outperforms all remaining categories with an ACC of 95.97%, followed by the symbol exploit category which achieves 82.03% in ACC. The webpage URL properties category is slightly inferior with an ACC of 76.13%. The less effective categories are the domain obfuscation category and the HTML content category, where both attained a rather low TN and TP rates of 49.93% and 58%, respectively. The imbalance performance of the domain obfuscation category and the HTML content category implies that these categories cannot reliably distinguish between phishing and legitimate webpages. Overall, we consider the content URL obfuscation category as the winning feature set, since it performs excellently with high TP and TN rates, which is crucial to correctly classify both classes (i.e., phishing and legitimate) of webpage. This key advantage is highly desirable in anti-phishing applications.

Since the performance of content URL obfuscation category is exceptionally good, we analysed further to investigate why the aforementioned feature category is better in differentiating between phishing and legitimate websites. The category with highest accuracy (i.e., content URL obfuscation) and the category with lowest accuracy (i.e., domain obfuscation) are analysed in terms of their feature values distribution. Fig. 1 and 2 show some examples of features from the aforementioned categories, where red and green colours represent feature values occupied by phishing and legitimate samples, respectively. The distribution of values along the x-axis provides an indication of the feature value type. For example, *NumDashInHostname* has values distributed in the x-axis range from 0 to 9, which represents a discrete value type. On the other hand, *IpAddress* has values distributed in the x-axis at either 0 or 1, which denotes a binary value type.

Based on Fig. 1, it can be observed that the feature values in content URL obfuscation category are more concentrated towards a certain range, especially for phishing samples. For example, we can easily deduce that a webpage with a larger value of *PercentExternalHyperlinks* is potentially a phishing webpage. On the other hand, the feature value distribution is less distinct as shown in Fig. 2, making it more difficult to associate a webpage sample with either phishing or legitimate webpage. From a computational point of view, a feature category with more distinctive patterns of feature value distribution is more superior, which could facilitate the classification algorithm to achieve higher detection accuracy. In addition, a feature category that performs better may indicate that such category contains more features that are consistently targeted by the current phishing schemes.

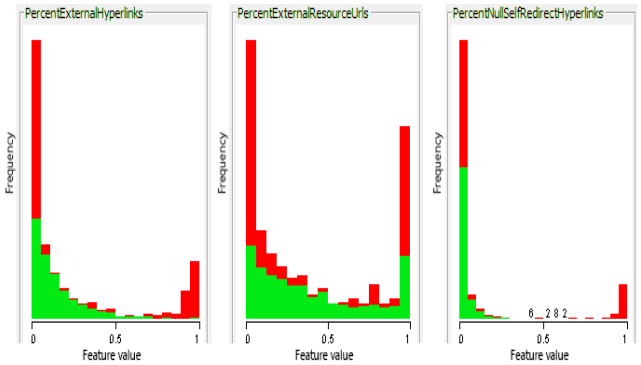

Fig. 1: Feature value distribution for content URL obfuscation category

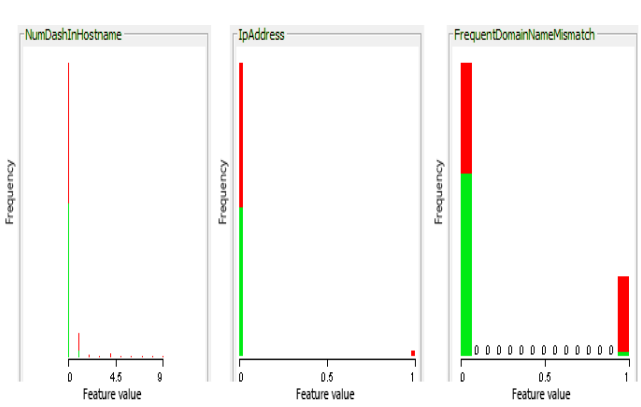

Fig. 2: Feature value distribution for domain obfuscation category

4.4. Evaluation II — Performance Comparison with Existing Anti-Phishing Techniques

In this evaluation, we generate two additional feature sets that have been utilised in He et al. [6] and Abdelhamid et al. [13] and used it to benchmark against our categorised winning feature set, namely the content URL obfuscation set. Note that the additional feature sets are using a combination of several feature categories.

Table 7: Performance metrics of winning feature set against other feature sets proposed in [6] and [13]

Feature Category	TP (%)	FN (%)	TN (%)	FP (%)	ACC (%)
Content URL obfuscation (winning feature set)	95.27	4.73	96.67	3.33	95.97
He et al. [6]	95.67	4.33	96.60	3.40	96.13
Abdelhamid et al. [13]	91.33	8.67	95.00	5.00	93.17

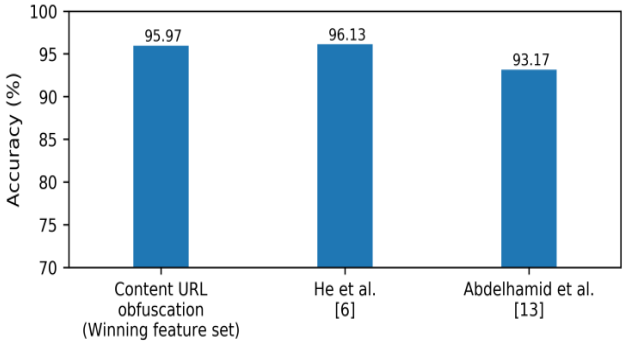

Fig. 3: Performance comparison of winning feature set against other feature sets proposed in [6] and [13]

Results in Table 7 and Fig. 3 suggest that the feature set of He et al. [6] achieved comparable performance to our categorised winning feature set. We found that the feature set of [6] contained 4 out of 7 features from our winning feature set, thus inferring that a majority of its correct predictions are attributed to the good features that came from our categorised winning feature set. The feature set of Abdelhamid et al. [13] achieved slightly lower accuracy at 93.17%, and we also found that their feature set contained 5 out of 7 features from our winning feature set. Therefore, our overall analysis suggests that existing phishing detection techniques tend to achieve high accuracy if a majority of features from our categorised winning feature set is being adopted in their classification model. As such, reliable computational-based methods such as filter measures are highly desirable for the purpose of features effectiveness evaluation, which will be explored in our future works.

5. Conclusion

In this work, an improved categorisation approach is introduced to categorise phishing detection features into precise categories. A large number of features were surveyed from existing phishing detection works and grouped according to the improved categorisation approach. Each feature category is evaluated using the C4.5 classifier, where the content URL obfuscation category has emerged as the winning set, outperforming the remaining categories. In addition, we conducted benchmarking to compare the performance of the winning feature set against other feature sets utilised in existing phishing detection techniques. Results suggest that the winning feature set is a crucial feature category for boosting the classification accuracy of machine learning-based phishing detection systems. The practical implications of this research include (a) enhancing the detection accuracy of machine learning-based phishing detection systems; (b) providing benchmark results to compare between different phishing feature categories, and; (c) identifying a small set of highly effective phishing features to minimise computational processing power.

In future, we plan to explore computational methods such as filter measures as a reliable approach to evaluate features and improve the phishing detection accuracy. In addition, we may also focus on optimisation of feature selection techniques to reduce the feature dimensionality and improve the efficiency of phishing classification methods. We also intend to further evaluate the features' performance using 10-fold cross validation method.

Acknowledgement

The funding for this project is made possible through the research grant obtained from UNIMAS under the Dana Pelajar PhD (Grant No: F08/DPP/1649/2018). This work is also supported by the Sarawak Foundation Tun Taib Scholarship Scheme.

References

[1] Anti-Phishing Working Group (2017), "Phishing Activity Trends Report, 1st Half 2017", available online: http://docs.apwg.org/reports/apwg_trends_report_h1_2017.pdf, last visit: 06.01.2018

[2] Bleau H (2016), "2017 Global Fraud and Cybercrime Forecast", available online: https://www.rsa.com/en-us/blog/2016-12/2017-global-fraud-cybercrime-forecast, last visit: 09.01.2017

[3] Purkait S (2015), "Examining the effectiveness of phishing filters against DNS based phishing attacks," *Information and Computer Security*, Vol. 23, No. 3, pp. 333–346.

[4] Varshney G, Misra M, & Atrey PK (2016), "A survey and classification of web phishing detection schemes," *Security and Communication Networks*, 2016.

[5] Gu X, Wang H, & Ni T (2013), "An efficient approach to detecting phishing web," *Journal of Computational Information Systems*, Vol. 9, No. 14.

[6] He M, Horng SJ, Fan P, Khan MK, Run RS, Lai JL, Chen RJ, & Sutanto A (2011), "An efficient phishing webpage detector," *Expert Systems with Applications*, Vol. 38, No. 10, pp. 12018–12027.

[7] Choo XM, Chiew KL, Ibrahim DHA, Musa N, Sze SN, & Tiong WK (2016), "Feature-based phishing detection technique," *Journal of Theoretical and Applied Information Technology*, Vol. 91, No. 1, pp. 101–106.

[8] Nguyen HH, & Nguyen DT (2016), "Machine learning based phishing web sites detection," *Proceedings of the International Conference on Advanced Engineering Theory and Applications (AETA)*, Ho Chi Minh City, Vietnam, pp. 123–131.

[9] Moghimi M, & Varjani AY (2016), "New rule-based phishing detection method," *Expert Systems with Applications*, Vol. 53, pp. 231–242.

[10] Garera S, Provos N, Chew M, & Rubin AD (2007), "A Framework for Detection and Measurement of Phishing Attacks," *Proceedings of the ACM Workshop on Recurring Malcode*, Alexandria, USA, pp. 1–8.

[11] Xiang G, Hong J, Rose CP, & Cranor L (2011), "CANTINA+: A Feature-rich Machine Learning Framework for Detecting Phishing Web Sites," *ACM Transactions on Information and System Security*, Vol. 14, No. 2, p. 21.

[12] Zhang Y, Hong JI, and Cranor LF (2007), "CANTINA: A Content-Based Approach to Detecting Phishing Web Sites," *Proceedings of the 16th International World Wide Web Conference*, Banff, Canada, pp. 639–648.

[13] Abdelhamid N, Ayesh A, & Thabtah F (2014), "Phishing detection based Associative Classification data mining," *Expert Systems with Applications*, Vol. 41, No. 13, pp. 5948–5959.

[14] Mohammad RM, Thabtah F, & McCluskey L (2012), "An assessment of features related to phishing websites using an automated technique," *Proceedings of the International Conference for Internet Technology and Secured Transactions*, London, UK, pp. 492–497.

[15] Ramesh G, & Krishnamurthi I (2014), "A comprehensive and efficacious architecture for detecting phishing webpages," *Computers & Security*, Vol. 40, pp. 23–37.

[16] Zuhair H, Selamat A, & Salleh M (2016), "New hybrid features for phish website prediction," *International Journal of Advances in Soft Computing and its Applications*, Vol. 8, No. 1, pp. 28–43.

[17] Schwartz B (2016), "Google has confirmed it is removing Toolbar PageRank", available online: https://searchengineland.com/google-has-confirmed-they-are-removing-toolbar-pagerank-244230, last visit: 27.05.2018.
[18] Sunil ANV, & Sardana A (2012), "A PageRank Based Detection Technique for Phishing Web Sites," *Proceedings of the IEEE Symposium on Computers and Informatics (ISCI)*, Penang, Malaysia, pp. 58–63.
[19] Mohammad RM, Thabtah F, & McCluskey L (2015), "Phishing Website Features," unpublished.
[20] Whittaker C, Ryner B, & Nazif M (2010), "Large-Scale Automatic Classification of Phishing Pages," *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, San Diego, USA.
[21] Ludl C, McAllister S, Kirda E, & Kruegel C (2007), "On the Effectiveness of Techniques to Detect Phishing Sites," *Proceedings of the 4th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA)*, Lucerne, Switzerland, pp. 20–39.
[22] Pan Y, & Ding X (2006), "Anomaly based web phishing page detection," *Proceedings of the 22nd Annual Computer Security Applications Conference (ACSAC)*, Miami Beach, USA, pp. 381–392.
[23] Fahmy HMA, & Ghoneim S (2011), "PhishBlock: A hybrid anti-phishing tool," *Proceedings of the International Conference on Communications, Computing and Control Applications (CCCA)*, Hammamet, Tunisia, pp. 1–5.
[24] Gupta S, & Kumaraguru P (2014), "Emerging Phishing Trends and Effectiveness of the Anti-Phishing Landing Page," *ArXiv e-prints*.
[25] Thabtah F, & Abdelhamid N (2016), "Deriving Correlated Sets of Website Features for Phishing Detection: A Computational Intelligence Approach," *Journal of Information & Knowledge Management*, Vol. 15, No. 4.
[26] Zuhair H, Selamat A, & Salleh M (2015), "The Effect of Feature Selection on Phish Website Detection," *International Journal of Advanced Computer Science and Applications*, Vol. 6, No. 10, pp. 221–232.
[27] Zuhair H, Selamat A, & Salleh M (2015), "Selection of robust feature subsets for phish webpage prediction using maximum relevance and minimum redundancy criterion," *Journal of Theoretical and Applied Information Technology*, Vol. 81, No. 2, pp. 188–205.
[28] PhishTank (2017), "Join the fight against phishing", available online: https://www.phishtank.com/, last visit: 10.01.2017.
[29] OpenPhish (2017), "Phishing Intelligence", available online: https://www.openphish.com/, last visit: 01.01.2017.
[30] Alexa Internet Inc. (2017), "Keyword Research, Competitive Analysis, & Website Ranking", available online: https://www.alexa.com/, last visit: 10.01.2017.
[31] "Common Crawl", available online: http://commoncrawl.org/, last visit: 10.01.2017.
[32] Selenium Project (2017), "Selenium WebDriver", available online: http://www.seleniumhq.org/projects/webdriver/, last visit: 10.01.2017.
[33] Frank E, Hall MA, & Witten IH (2016), The WEKA Workbench, 4th edn. *Morgan Kaufmann*, Burlington, Massachusetts, pp. 553–571.