Malicious URL Detection : A Survey

Eint Sandi Aung[†a)] Hayato YAMANA[†b)]

†Department of Computer Science and Communication Engineering, Graduate School of Fundamental Science and Engineering, Waseda University, Tokyo, 169-8555, Japan.

E-mail : a) eintsandiaung@toki.waseda.jp, b) yamana@info.waseda.ac.jp

Abstract Phishing is a type of fraud, in which two actors, phisher and victim, take part. The role of a phisher is to create a phishing webpage by mimicking as an authorized one and embed the website in an email or any other media. A victim may access to the phished link without consciousness or with lack of knowledge. Detecting malicious URLs (Uniform Resource Locators) is a challenging, yet interesting topic because phishers mostly generate URLs randomly and researchers have to detect them while considering the behaviors behind the generated phishing URLs. Although various detection schemes exist in anti-phishing area, URL-based scheme is safer and more realistic because of two perspectives: no need of access to the malicious webpage and an ability of zero-hour detection. Therefore, in our paper, we survey malicious URL detection by approaching a variety of existing technical countermeasures and analyze existing weakness. Eventually, we conclude our survey with potential opinion for more effective detection.

Keyword Malicious URL, Phishing, Security

1. Introduction

In recent years, information security has become a trendy subject since many people have suffered from leakage of personnel information. At the same time, attackers try to impersonate as an authorized person or organization. They use any form of medium to attract users, such as adding persuasive ads or pop-ups in social network services, embedding fake links in emails or compromising an authentic website. Such frauds are known as phishing. To be simply defined, phishing is a type of threat of personnel information where phishers intentionally attack a person or organization.

APWG[1] reported that the number of phishing attacks reached its peak with approximately 250,000 in Q3 of 2019 within three-years period. Moreover, it was reported that forty percent of business email compromise (BEC) attacks utilized domain names registered by the criminals. They created similar domain names of trusted existing company names to bait gullible users. Fifty-four percent of BEC attacks used free webmail in the Q3. In addition, around 66% of of all phishing sites reported to APWG used SSL protection, which was the highest percentage since 2015, indicating that users cannot absolutely rely on SSL. These reports claim that URLs have been a vector to be deceived by phishers since common users are not fully attentive to suspicious URLs.

Our work aims to survey a varying trend of malicious URL detection and to analyze a variety of detection techniques changing over time.

We organize this paper into five different sections: Section 2 describes the background of phishing respective to detection perspectives. Then, Section 3 lists various detection methods over time, followed by Section 4 summarizing our discussion. Eventually, Section 5 concludes our survey paper.

2. Background

In the literacy of phishing detection, there is a variety of detection perspectives. Phishing attack can occur through different vectors, which we discuss in section 2.1. Then, we describe different perspectives in the following section 2.2.

2.1. Phishing Attack Vector

Since phishers always search for any possible way to attack ordinary users, attack vectors are different every time. However, we can define the common media as web, mail and network.

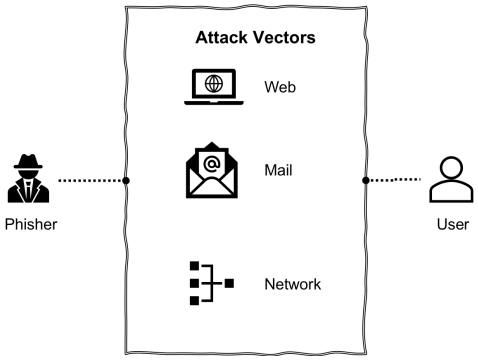

Figure 2.1: Phishing Attack Vector

2.1.1. Web

Web vector can be used when phishers compromise a legitimate website. They often target user's unawareness of checking URL and make them trust on the compromised web link. Normally, web link is embedded in a phishing mail. In addition, phishers also use another technique, i.e., they create seem-to-be-realistic URLs which has visually similar name with legitimate website. When a user types in some typo errors, they are directed to the phishing website.

2.1.2. Mail

Mail vector can be used for two different targets such as individual or organization. Phishers target superiors in an organization and trick them with fake trust-worthy information. Then, phishers send enormous emails to a vast number of individuals by luring as if they are reliable sources. Normally, individual attack occurs after they successfully break into an organization's network.

2.1.3. Network

Phishing attack through network can generally happen when router is hijacked by exploiting a vulnerability into the router's firmware. After attackers taking over the router, they change addresses of DNS server the router uses to resolve domain names.

2.2. Detection Perspectives

Categorizing detection techniques can vary in terms of research focus. However, in our work, we approach the detection with two perspectives, fundamentally. We can categorize them into (1)database- and (2)heuristic-oriented perspectives as shown in figure 2.2. We describe details in the rest of the section.

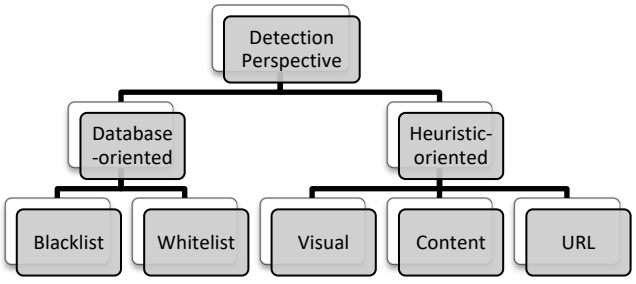

Figure 2.2: Phishing Detection Perspectives

2.2.1. Database-oriented detection

Most of the phishing detection systems use database-oriented detections, traditional approaches, such as blacklist and whitelist. These approaches are common in earlier detection systems since they can give faster output and more convenient detection. However, both of them have several limitations, i.e., in blacklist approach, as long as there is no prior similar information, then result could be highly false positive [29]. Similarly, in whitelist approach, there can be misclassification even when a user is authentic if the user accesses to unfamiliar website because of no history of often-accessed links [30].

2.2.2. Heuristic-oriented detection

Heuristic-oriented detection varies from content to visual-based detection. Basically, heuristic-oriented detection results more precise performance in terms of accuracy, precision and recall. They are more robust than database-oriented detection. They are visual, content and URL-based techniques. However, there are also a few drawbacks in these techniques. Since visual detection checks if the two websites are visually similar, it consumes a longer execution time leading to be unrealistic [31][32]. In content-based detection such as extracting keywords using tf-idf (term frequency – inverse document frequency) could give wrong detection result when phishers rarely use texts in a webpage. Then, it might lead to high false positive values [27]. Furthermore, although URL-based technique can detect more accurately, it highly relies on features used in the system. The more the number of features, the better accuracy, however, the longer training time [5] and unnecessary features could even lead to reduce performance.

2.3. Comparison of detection techniques

Phishing detection is an interesting yet challenging topic in security area as phishers always overcome with an innovative technique to sneak into a system. There is no perfect system in phishing detection area. Researchers come up to diverse approaches depending on distinct criteria, however, each solution has its own limitation. We illustrate the limitations in table 2.1.

Table 2.1: Comparison of detection techniques

Detection Perspective		Web Access	Zero-hour Detection	Processing Time	Accuracy
Database-oriented	**Blacklist**	Depend	No	Depend	Low
	Whitelist	Depend	No	Depend	Low
Heuristic-oriented	**Visual**	Yes	No	Long	High
	Content	Yes	No	Long	High
	URL	No	Yes	Short	High

In terms of web access, it might be vulnerability because of the possibility that phishers install a malware into a system. Thus, visual ad content-based perspectives can be at risk. In zero-hour aka real-time detection, database- and two of the heuristic-oriented approaches, such as visual and content, cannot perform well since they need to access to database and the whole content of a webpage, respectively. Moreover, visual and content approaches also take longer processing time than the rest while database-oriented ones result lower accuracy than the other. Among all of them, URL detection technique is at low risk in terms of malware installation, highly performs in zero-hour detection i.e. for newly created websites, processes faster and results higher accuracy, however, it increases false positive rates if features are not extracted properly.

3. Malicious URLs and Detection Methods

In our work, we analyze malicious URLs with respect to different detection methods. Methods used in phishing detection field have been changing over time as researchers keep working on to build a more robust detection system and to overcome advanced attacks by phishers. Furthermore, there were various surveys in phishing area. However, they discussed about different phishing perspectives and a limited number of them focused on URL [3][7][25]. In our work, we refer URL as the only resource with no need of access to the website. We solely focus on URL-related attack and survey its various methods. To be noted that although previous works used different methods such as weighting-[24], rule-[23], machine learning- and neural network-based, we discuss machine learning- and neural network-based methods in our survey.

3.1. Machine Learning Algorithms

Since the past few decades, machine learning has been essential in data science field. In phishing area, most of the detection techniques apply machine learning algorithms until now. Thus, we list some of them with respective previous works in table 3.1.

3.1.1. Naïve Bayes

Naïve Bayes classifier is a generative probabilistic model in machine learning and is based on the Bayes theorem. It is mostly used in classification areas, such as text classification, spam detection, because of its simplicity. Its features are independent among each other. The conditional probability of Naïve Bayes classifier is described as follows.

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)} \quad (1)$$

Where P(B) is the prior probability of B being true, which is B={0,1} while P(A) is the evidence or a set of feature vectors. P(B|A) is the posterior probability in which B is true given A. P(A|B) is the probability of A given B being true.

Table 3.1: Machine Learning Algorithms

Machine Learning Algorithms	**Traditional**		Naïve Bayes	[1][2][5][6]
			Support Vector Machine (SVM)	[1][3][4][5][6][7][23][25]
			kNearest Neighbor (kNN)	[1][2][6][7][8]
	Tree-based	**Decision Trees**	Decision Tree	[1][2][6][8]
			ID3	[2]
			J48	[5][7][9]
			ADTree	[9]
			Random Tree	[9][23]
			CART	[9]
			REPTree	[9]
		Ensemble	Random Forest	[1][2][5][7][8][9][10][23]
			Gradient Boosting	[6][8]
			Adaboost	[1]
			XGBoost	[5][8]
			Deep Forest	[8]
			Majority Voting	[6][9]
	Regression		Logistic Regression	[5][8]

3.1.2. Support Vector Machine (SVM)

Support vector machine can solve linear or non-linear problems. In linear problems, it simply finds a hyperplane in N-dimensional feature space. However, in non-linearly separable problems, SVM uses kernel trick in training dataset [28][25]. Sequential Minimal Optimization (SMO) is a fast learning method for SVM and also used in Weka. SVM minimizes loss by maximizing the margin between boundary and data point.

$$(w, b) = arg_{w,b} \min \frac{1}{T} \sum_{t=1}^{T} \max\left(0, 1 - y_t(w.x_t + b)\right) + \lambda \|w\|_2^2 \quad (2)$$

3.1.3. kNearest Neighbor (kNN)

kNN is a non-parametric algorithm used in both classification and regression. Its classification works on unknown data closest to k in the training feature space. Closest points are selected using distance functions such as Hamming, Euclidian and Minkowski. kNN works slow if the data size is large.

3.1.4. Decision Trees

Decision tree classifiers are one of the most popular classifiers used in classification and regression. It divides

the training dataset until it reaches to a leaf node, which is a label in classification. Decision tree classifier uses the entire training dataset while constructing a tree unlike Random Forest. Some of the popular decision tree classifiers are CART, random tree, J48, ADTree and REPTree.

3.1.5. Random Forest

Random Forest is an ensemble classifier used for classification and regression. It constructs decision trees based on randomly selected sets in training samples and then aggregate decisions from these trees by averaging or majority voting. It improves accuracy and also reduces overfitting.

3.1.6. Gradient Boosting

Gradient boosting is one of the popular boosting in ensemble learning. As boosting models learn from previous mistakes, gradient boosting learns from residual error directly unlike AdaBoost which updates the weights of data points. It is a generic algorithm to find approximate solutions.

3.1.7. AdaBoost

AdaBoost (Adaptive Boosting) works as a conjunction algorithm because it is used to classify by training different weak learning algorithms to form a strong one i.e. to improve performance. The output of weak classifiers are combined by setting correct weights for final decision. Since AdaBoost is sensitive to outliers and focuses on hard-to-classify samples, it is less resistant to overfitting.

3.1.8. XGBoost

XGBoost (Extreme Gradient Boosting) is a specific form of gradient boosting methods and uses more accurate approximation for the best decision tree. It computes second partial derivatives of the loss function and performs advanced regularization. It has the advantage of fast training time and training process can be distributed.

3.1.9. Logistic Regression

Logistic regression is a discriminative probabilistic model mainly used in which the output is binary. Logistic regression performs better than Naïve Bayes model when training size is close to infinity.

$$P(B|A) = \frac{1}{1+e^{\beta_0+\sum_i^n \beta_i A_i}} \text{ if B=1}$$

$$P(B|A) = \frac{e^{\beta_0+\sum_i^n \beta_i A_i}}{1+e^{\beta_0+\sum_i^n \beta_i A_i}} \text{ otherwise} \quad (3)$$

3.2. Neural Network Algorithms

Since few years ago, neural network has become popular in data science because of its outstanding accuracy. However, such learning algorithms are mostly used in computer vision-related area (e.g. image classification). Previously in text mining, it was rarely used but recently most of the text mining-related researches apply neural network algorithms by encoding texts and embedding them to fit into the algorithms criteria. We describe a brief insight into some artificial neural network learning models. Existing works related to basic neural network can be seen in [11][19].

3.2.1. Convolutional Neural Network (CNN)

Convolutional neural network is widely used in computer vision such as face recognition. It is similar to the basic neural network i.e. it has learnable parameters: weights and bias. It is a deep feed-forward neural network. CNN extracts features of text, here, and converts it into lower dimension maintaining its characteristics. Connections in CNN between nodes do not form a circle. Existing works can be seen in [12][14][15][20].

3.2.2. Recurrent Neural Network (RNN)

Recurrent neural network (RNN) is a class in artificial neural network in which connections between nodes form a directed graph with its sequence. RNN overcomes traditional neural network weakness such as a problem handling with sequence data (e.g. text and time series)[12]. Long Short Term Memory (LSTM) network is a special design of recurrent neural network, designed to overcome long-term dependency problem. RNN-related works can be seen in [13][14][16][18][21].

3.2.3. Generative Adversarial Network (GAN)

Generative adversarial network is a generative model which tries to copy the given data and generate looked-alike texts or images. It has two parties, generator and discriminator, competing against each other to form a robust model. Recently, researchers became interested in GAN for text generating of phishing URLs.[17][18]

3.2.4. Neural Network Architecture

We discuss the most common architecture used in malicious URL detection in artificial neural network. Three actors in neural network architecture are input, neuron and output.

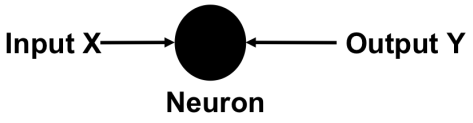

Figure 3.1: Participants in Neural Network

In hidden layer, neural network considers two parameters, bias and weight and computes activation function (e.g. sigmoid, tanh and ReLU). Such parameters can be updated by reducing loss function. Hidden layers can be single or

multi layers.

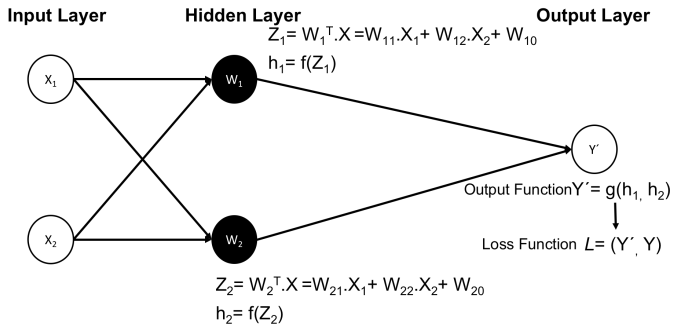

Figure 3.2: Neural Network Layers

4. Detection Analysis

We filtered out previous works only related to malicious URLs. We performed comprehensive survey of various research focus because each research has individual concept to approach the problems in that period of time. As narrowing down to URL-based phishing detection alone has limitations on the number of previous researches, we mention a couple of previous work which used hybrid methods such as URL- and content-based. Our survey process is described in table 4.1.

Table 4.1: Filtration Process

Search Engine	Google Scholar	
Keywords	・phishing ・URL ・detection ・survey ・malicious ・generate	Combinations using AND operator in search engine: ・phishing AND URL AND detection, ・phishing AND detection ・phishing AND detection AND survey ・malicious AND URL AND detection ・generative AND adversarial AND network AND malicious AND URL
Year	2017~2019 [primary] 2014~2016 [secondary]	Primary: High priority selection Secondary: Low priority selection (used mostly when to eliminate similar work)
Filtered Paper Focus	URL-based detection	Manually filtered out based on abstract of paper
Total No. of Papers	26 [Ref.1-26]	

After we collected 26 papers, we eliminated papers based on two aspects: (1)when the paper merely performed comparisons of different classifiers, (2)when no novelty of method in the paper. We dropped out 13 papers in total from the list. However, as those papers follow filtration process in Table 4.1 although they have no novelty, we list their applied yet existing methods in previous Section 3. Eventually, we list papers which have contributions in terms of framework, feature engineering or algorithm and list them in Table 4.2.

4.1. Discussion of Detection Trends

After analyzing detection trends varying over time, we found that until 2018, researchers focused on classification using various machine learning techniques. Their objective was mainly on detection. However, since 2018, the trend switched to a more comprehensive and interesting approach i.e. generating potential/candidate URLs by using neural network based methods as the objective of research is to expose malicious URLs beforehand.

Moreover, we noticed that neural networks-related methods are gradually being used in phishing URL detection area. Although these methods are mostly used in computer vision, recently they are applied on text data after transforming the data into metrics.

5. Conclusion

In this survey, we first defined phishing detection perspectives based on the necessity of database access or not. We named them as database-oriented and heuristic-oriented perspectives. In the later section of the paper, we approached existing methods and grouped them as machine learning- and neural network-based. Then, we described previous works having different perspectives approaching to the problems. We eliminated some of the overlapping works with no contribution. Eventually, we concluded our survey with detection trend then and now.

References

[1] Sahingoz, E. Buber, O. Demir, and B. Diri, “Machine learning based phishing detection from URLs,” in Expert Systems with Applications, vol.117, pp.345-357, March,2019. DOI:10.1016/j.eswa.2018.09.029

[2] M. T. Suleman, and S. M. Awan, “Optimization of URL-based phishing websites detection through genetic algorithms,” in Automatic Control and Computer Sciences, vol.53, no.4, pp.333-341, September, 2019. DOI:10.3103/s0146411619040102

[3] D. Sahoo, C. Liu, and S. C. H. Hoi, “Malicious URL detection using machine learning : a survey,” In ArXiv, vol.abs/1701.07179v3, 2019.

[4] R. Patgiri, H. Katari, R. Kumar, and D. Sharma, “Empirical study on malicious URL detection using machine learning,” Proc. 15th International Conference on Distributed Computing and Internet Technology, Bhubaneswar, India, pp.380-388, January, 2019. DOI:10.1007/978-3-030-05366-6-31

[5] C. Liu, L. Wang, B. Lang, and Y. Zhou, “Finding effective classifier for malicious URL detection” Proc. 2nd International Conference on Management Engineering, Software Engineering and Service Sciences, pp.240-244, January, 2018. DOI:10.1145/3180374.3181352

Table 4.2: Comprehensive Analysis of Malicious URL Detection

No	Title	Research Description	Pros	Cons/Limitations	Performance Evaluation	Year
1	Phishing URL Detection Via CNN And Attention-Based Hierarchical RNN [12]	· Proposed **PhishingNet** (Deep Learning Approach) · Used CNN for character-level feature extraction · Used **Attention-based hierarchical RNN** for word-level feature extraction · **Fused character and word-level features via CNN**	· Improved generalization ability on newly emerging URLs	· Took longer execution time while fusing CNN and RNN	AUC:0.9926 ACC:0.9791 FPR:0.0002 Precision:0.9896	2019
2	Phishing URL Detection Via Capsule-Based Neural Network [11]	· Proposed **Capsule-based** neural network · **Primary capsule layer**: extracted accurate features from shallow features generated by former convolution layer and utilized batch normalization · **Classification capsule layer**: used dynamic routing algorithm and squashing function and averaged outputs from all branches	· Parallel branches make effective for extensive experiments	· Lower true positive rate (0.9349) than URLNet (0.9933)	AUC:0.9966 ACC:0.9963 FPR:0.0005 Precision:0.9898 TPR:0.9349 Recall:0.0349 F1:0.9616	2019
3	Machine Learning Based Phishing Detection From URLs [1]	· Proposed real-time anti-phishing system · Used seven different classification algorithms and natural language processing (NLP) based features	· Independence of third-party services · **Language independence** · **Real-time execution**	· Cannot detect URL with only single domain name (e.g.www.testbank.com) due to NLP based features	AUC:0.9798 Precision:0.9700 Sensitivity:0.9900 FMeasure:0.9800	2019
4	Phishing URL Detection With Oversampling Based On Text Generative Adversarial Networks [18]	· Proposed **oversampling technique of URLs** and used **text-GAN** in minority class for data space · Performed oversampling using conventional features for feature space	· Showed possibility of hunting phishing URLs by prior generating 40~80 of them	· Use of 4 datasets with relatively small size could be incomprehensive	(ebay) AUCROC:0.7010 F1:0.6991 F2:0.6860 (PayPal) AUCROC:0.7101 F1:0.6993 F2:0.8100 (Bank of America) AUCROC:0.7010 F1:0.7012 F2:0.7807 (Sorio et al.) AUCROC:0.9765 F1:0.9738 F2:0.9698	2018
5	Robust URL Classification With Generative Adversarial Networks [17]	· Used **GAN for URL classification** · Used datasets of log files collected by Tstat[33]	· **Real datasets collected from log** files · Highly correctly classified for three benign-datasets	· Poorly classified for malware dataset	Benign datasets (Checkpoint) precision:0.9900 (Video) Precision:0.9900 (Windows) Precision:0.9600 Malware dataset (Tidserv) Precision:0.5600	2018
6	DeepPhish: Simulating Malicious AI [16]	· **Identified threat actors** and proposed **DeepPhish algorithm** to demonstrates potential attack used by threat actors · Applied LSTM network	· Improved effectiveness rate* in each threat actors *Effectiveness rate is measured by number of URLs bypassed	· Unable to model success because of data limitation *Success rate is measured by number of URLs which	(Threat actor 1) Effectiveness:20.90% (Threat actor 2) Effectiveness:36.28%	2018

			detection system over the total generated URLs with same technique	actually stole user information		
7	Acquire, Adopt, And Anticipate: Continuous Learning To Block Malicious Domains [15]	· Proposed **automated learning system** · Develops deep learning model · Publishes unreported malicious domains · **Periodically updates** detection models ·	· Anticipated domains similar to known malicious domains · Did not generate known legitimate domain	· Failed to put common keywords together	Ration of domains blacklisted after being detected by system:9.36%	2018
8	Malicious Domain Name Detection Based On Extreme Machine Learning [19]	· Proposed machine learning based methodology using **Extreme Learning Machine (ELM)** for malicious domain detection · Used Single-hidden-Layer-Feedforward networks (SLFNs) and moduled detection problem as SLFN	· Fast learning speed	· Detection rate and accuracy dropped if no of nodes are larger than 1000	@500 nodes Detection rate: 0.9427 ACC:0.9629	2018
9	URLNet: Learning A URL Representation With Deep Learning For Malicious URL Detection [20]	· Proposed **end-to-end deep learning framework, URLNet** · Learn nonlinear embedding directly from URLs · Applied CNN for both character- and word-level embedding	· Able to learn unseen words Able to detect sequential words	Took longer execution time	AUC:0.9929	2017
10	Phishing webpage detection using weighted URL tokens for identity keywords retrieval [26]	· Proposed anti-phishing technique using **weighted URL tokens** · Extract identity keywords from a quried webpage	· Worked well on non-English webpage and outperforms CANTINA [27] ·	· Included third-party services (DNS lookup) · Language dependency ·	ACC:0.9570	2017
11	New Rule-Based Phishing Detection Method [22]	· Proposed features independent from third-party services · **Embedded** extracted rules into **browser extension**	· Embedding extracted rules into browser extension makes the detection faster	· Features entirely dependent on webpage content · Incorrectly detection if attackers do not use DOM	TPR:0.9914 FNR:0.0860	2016
12	PhishStorm: Detection Phishing With Streaming Analytics [23]	· Proposed PhishStorm, a real-time automated detection system · Proposed a **new intra-URL relatedness**	· Quickly request the local database to compute intra-URL relatedness	· Not applicable to all types of obfuscated URLs · Limited publicly available data from Google Trends and Yahoo Clues	ACC:0.9491 Precision:0.9844 FMeasure:0.9472	2014
13	Phishing Website Detection Using URL-Assisted Brand Name Weighting System [24]	· Proposed a detection approach to find legitimate **domain names which use brand names** · Assign weights to words extracted from URLs	· Effectively extracted brand names using TF-IDF	· Dependent on search engines · Phishing websites hosted on free hosting servers caused FP rate · Relatively low dataset	ACC:0.9725 TPR:0.9820 FPR:0.0588	2014

[6] H. Shirazi, B. Bezawada, and I. Ray, "Know thy doma1n name: unbiased phishing detection using domain name based features," Proc. the 23nd ACM on Symposium on Access Control models and Technologies, pp.69-75, June, 2018. DOI:10.1145/3205977.3205992

[7] W. Daffa, O. Bamasag, and A. AlMansour, "A survey on spam URLs detection in Twitter," Proc. 1st International Conference on Computer Applications and Information Security, Riyadh, Saudi Arabia, pp.1-6, April, 2018. DOI:10.1109/CAIS.2018.8441975

[8] H. Yuan, X. Chen, Y. Li, Z. Yang, and W. Liu, "Detecting phishing websites and targets based on URLs and webpage links," Proc. 24th International Conference on Pattern Recognition, Beijing, China, pp.3669-3674, August, 2018. DOI:10.1109/ICPR.2018.8546262

[9] D. Patil, and J. Patil, “Malicious URL detection using decision tree classifiers and majority voting technique,” in Cybernetics and Information Technologies, vol.18, Issue.1, pp.11-29, March.2018. DOI:10.2478/cait-2018-0002

[10] S. Parekh, D. Parikh, S. Kotak, and S. Sankhe, “A new method for detection of phishing websites: URL detection,” Proc. 2nd International Conference on Inventive Communication and Computational Technologies, Coimbatore, India, pp.949-952, 2018. DOI:10.1109/ICICCT.2018.8473085

[11] Y. Huang, J. Qin, and W. Wen, “Phishing URL detection via capsule-based neural network,” Proc. 13th International Conference on Anti-counterfeiting, Security, and Identification, Xiamen, China, pp.22-26, October, 2019. DOI:10.1109/ICASID.2019.8925000

[12] Y. Huang, Q. Yang, J. Qin, and W. Wen, “Phishing URL detection via CNN and attention-based hierarchical RNN,” Proc. 18th International Conference on Trust, Security and Privacy in Computing and Communications and 13th International Conference on Big Data Science and Engineering, Rotorua, New Zealand, pp.112-119, August, 2019. DOI:10.1109/TrustCom/BigDataSE.2019.0024

[13] S. Shivangi, P. Debnath, K. Sajeevan, and D. Annapurna, “Chrome extension for malicious URLs detection in social media applications using artificial neural networks and long short term memory networks,” Proc. 18th International Conferences on Advances in Computing, Communications and Informatics, Bangalore, India, pp.1993-1997, September, 2018. DOI:10.1109/ICACCI.2018.8554647

[14] A. Vazhayil, R. Vinayakumar, and K. P. Soman, “Comparative study of the detection of malicious URLs using shallow and deep networks,” Proc. 9th International Conference on Computing, Communication and Networking Technologies, Bangalore, India, pp.1-6, July, 2018. DOI:10.1109/ICCCNT.2018.8494159

[15] I. Arnaldo, A. Arun, and S. Kyathanahalli, “Acquire, adapt and anticipate: continuous learning to block malicious domains,” Proc. International Conference on Big Data, Seattle, USA, December, 2018. DOI:10.1109/BigData.2018.8622197

[16] A. C. Bahnsen, I. Torroledo, D. Camacho, and S. Villegas, “DeepPhish: simulating malicious AI,” in APWG Symposium on Electronic Crime Research, 2018.

[17] M. Trivesan, and I. Drago, “Robust URL classification with generative adversarial networks,” in ACM SIGMETRICS Performance Evaluation Review, vol.46, no.3, pp. 143-146, December, 2018. DOI:10.1145/3308897.3308959

[18] A. Anand, K. Gorde, J. R. A. Moniz, N. Park, T. Chakraboty, and B. Chu, “Phishing URL detection with oversampling based on text generative adversarial networks,” Proc. International Conference on Big Data, Seattle, USA, pp.1168-1177, December, 2018.
DOI:10.1109/BigData.2018.8622547

[19] Y. Shi, G. Chen, and J. Li, “Malicious domain name detection based on extreme machine learning,” in Neural Processing Letters, vol.48, pp.1347-1357, 2018. DOI:10.1007/s11063-017-9666-7

[20] H. Le, Q. Pham, D. Sahoo, and S. C. H. Hoi, “URLNet: learning a URL representation with deep learning for malicious URL detection,” In ArXiv, vol.abs/1802.03162, 2017.

[21] A. C. Bahnsen, and E. C. Bohorquez, “Classifying phishing URLs using recurrent neural networks,” in APWG Symposium on Electronic Crime Research, pp.1-8, 2017. DOI:10.1109/ECRIME.2017.7945048

[22] M. Moghimi, and A. Y. Varjani, “New rule-based phishing detection method,” in Expert Systems with Applications: An International Journal, vol.53, pp.231-242, July, 2016. DOI:10.1016/j.eswa.2016.01.028

[23] S. Marchal, J. Francois, R. State, and T. Engel, “PhishStorm: detecting phishing with streaming analytics,” in IEEE Transactions on Network and Service Management, vol.11, no.4, pp.458-471, December, 2014.
DOI:10.1109/TNSM.2014.2377295

[24] C. L. Tan, K. Chiew, and S. Sze, “Phishing website detection using URL-assisted brand name weighting system,” in International Symposium on Intelligent Signal Processing and Communication Systems, pp.54-59, 2014.
DOI:10.1109/ISPACS.2014.7024424

[25] D. Sahoo, C. Liu, and S. C. H. Hoi, “Malicious URL detection using machine learning : a survey,” In ArXiv, vol.abs/1701.07179v1, 2017.

[26] C. L. Tan, K. L. Chiew, and S. N. Sze, “Phishing webpage detection using weighted URL tokens for identity keywords retrieval,” Proc. 9th International Conference on Robotic, Vision, Signal Processing and Power Applications, vol.398, pp.133-139, 2017. DOI:10.1007/978-981-10-1721-6

[27] Y. Hang, J. Hong, L. Cranor, “CANTINA: a content-based approach to detecting phishing web sites,” Proc. 16th International Conference on World Wide Web, pp.639-648, January, 2007. DOI:10.1145/1242572.1242659

[28] A. J. Smola, and B. Scholkopf, Learning with kernels, Citeseer, 1998.

[29] P. Prakash, M. Kumar, R. R. Kompella, and M. Gupta, “PhishNet: predictive blacklisting to detect phishing attacks,” Proc. IEEE INFOCOM, pp.1-5, March, 2010. DOI:10.1109/INFCOM.2010.5462216

[30] Y. Cao, W. Han, and Y. Le, “Anti-phishing based on automated individual white-list,” Proc. 4th ACM Workshop on Digital Identity Management, New York, USA, pp.51–60, 2008. DOI:10.1145/1456424.1456434

[31] E. Medvet, E. Kirda, and C. Kruegel. “Visual-similarity based phishing detection,” in Proceeding of 4th International Conference on Security and Privacy in Communication Networks, Article.22, pp.1-6, 2008.
DOI:10.1145/1460877.1460905

[32] Y. Fu, L. Wenyin, and X. Deng, "Detecting phishing web pages with visual similarity assessment based on earth mover's distance (EMD)," in IEEE Transactions on Dependable and Secure Computing, vol. 3, no. 4, pp.301-311, 2006.

[33] M. Trevisan, A. Finamore, M. Mellia, M. Munafo, and D. Rossi, “Traffic analysis with off-the-shelf hardware: challenges and lessons learned,” in IEEE Communications Magazine, vol.55, no.3, pp.163-169, 2017.