УНИВЕРСИТЕТ ИТМО

НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
март–апрель 2022 Том 22 № 2 http://ntv.ifmo.ru/
SCIENTIFIC AND TECHNICAL JOURNAL OF INFORMATION TECHNOLOGIES, MECHANICS AND OPTICS
March–April 2022 Vol. 22 No 2 http://ntv.ifmo.ru/en/
ISSN 2226-1494 (print) ISSN 2500-0373 (online)

НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК УНИВЕРСИТЕТ ИТМО
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
март–апрель 2022 Том 22 Номер 2

doi: 10.17586/2226-1494-2022-22-2-262-268

Lightweight approach for malicious domain detection using machine learning

Ganesan Pradeepa[1]✉, Radhakrishnan Devi[2]

[1,2] Vels Institute of Science, Technology and Advanced Studies, Pallavaram, Chennai, 600117, India

[1] pradeepa25.ganesan@gmail.com✉, https://orcid.org/0000-0002-5920-066X
[2] devi.scs@velsuniv.ac.in, https://orcid.org/0000-0002-8951-2242

Abstract
The web-based attacks use the vulnerabilities of the end users and their system and perform malicious activities such as stealing sensitive information, injecting malwares, redirecting to malicious sites without their knowledge. Malicious website links are spread through social media posts, emails and messages. The victim can be an individual or an organization and it creates huge money loss every year. Recent Internet Security report states that 83 % of systems in the internet are infected by the malware during the last 12 months due to the users who do not aware of the malicious URL (Uniform Resource Locators) and its impacts. There are some methods to detect and prevent the access malicious domain name in the internet. Blacklist-based approaches, heuristic-based methods, and machine/deep learning-based methods are the three categories. This study provides a machine learning-based lightweight solution to classify malicious domain names. Most of the existing research work is focused on increasing the number of features for better classification accuracy. But the proposed approach uses fewer number of features which include lexical, content based, bag of words, popularity features for malicious domain classification. Result of the experiment shows that the proposed approach performs better than the existing one.

Keywords
machine learning, lexical features, malicious domain, support vector, random forest, feature selection, cyber security

For citation: Pradeepa G., Devi R. Lightweight approach for malicious domain detection using machine learning. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2022, vol. 22, no. 2, pp. 262–268. doi: 10.17586/2226-1494-2022-22-2-262-268

УДК 004.89

Облегченный подход к обнаружению вредоносных доменов с использованием машинного обучения

Ганесан Прадипа[1]✉, Радхакришнан Деви[2]

[1,2] Институт науки, технологий и перспективных исследований Велса, Паллаварам, Ченнаи, 600117, Индия

[1] pradeepa25.ganesan@gmail.com✉, https://orcid.org/0000-0002-5920-066X
[2] devi.scs@velsuniv.ac.in, https://orcid.org/0000-0002-8951-2242

Аннотация
Веб-атаки используют уязвимости конечных пользователей и их систем. Атаки выполняют вредоносные действия, такие как кража конфиденциальной информации, внедрение вредоносных программ, перенаправление на вредоносные сайты без ведома пользователя. Вредоносные ссылки на веб-сайты распространяются через публикации в социальных сетях, электронные письма и сообщения. Жертвой может быть физическое лицо или организация, и каждый год такие действия приносят огромные денежные потери. В недавнем отчете Internet Security сказано, что 83 % систем в Интернете за последний год были заражены вредоносным программным обеспечением, так как пользователи не знали о воздействии вредоносного Uniform Resource Locator (URL)-адреса. Существует несколько способов обнаружения и предотвращения доступа к вредоносным доменным именам. Известные подходы основаны на черном списке, эвристических методах и методах, основанных на машинном глубоком обучении. В работе представлено облегченное решение классификации вредоносных доменных имен на основе машинного обучения. Большая часть существующих исследований направлена

на повышение точности классификации с помощью увеличения количества вредоносных признаков. В предложенном подходе использовано меньшее количество функций, включая лексические, основанные на содержании, наборе слов, популярных функциях для классификации вредоносных доменов. Результат эксперимента показал, что представленный подход работает лучше, чем существующие.

Ключевые слова
машинное обучение, лексические признаки, вредоносный домен, опорный вектор, случайный лес, выбор признаков, кибербезопасность

Ссылка для цитирования: Прадипа Г., Деви Р. Облегченный подход к обнаружению вредоносных доменов с использованием машинного обучения // Научно-технический вестник информационных технологий, механики и оптики. 2022. Т. 22, № 2. С. 262–268 (на англ. яз.). doi: 10.17586/2226-1494-2022-22-2-262-268

Introduction

Business, education, research, and the access to numerous essential services in our daily lives have all been altered by the internet and the World Wide Web. It removes all the communication barriers. While we get many benefits from the internet, it also provides ample space for illegal activities. These illegal activities include money laundering, personal information theft and malware installation, etc. The specially crafted websites for these activities, called malicious sites, and the Uniform Resource Locators (URLs) that refer to the sites, called as malicious URLs. The malicious website contains unsolicited content which invites the internet users to fall in the trap. Cyber criminals use every opportunity even in the pandemic periods to increase wide range of attacks and causing huge money loss. Recent cyber security reports of 2020 state that during the height of global epidemic fears, the number of cyber-attack incidents increased by a staggering 220 % compared to the annual average [1]. It is necessary to prevent the access of the malicious URLs to safeguard the internet users.

Many techniques were proposed by the researchers [2, 3] based on different techniques such as blacklisted URL technique (huge list of malicious URLs are collected and blocked from usage), heuristic technique (frames of the generalized rules based on the dataset of URLs for detecting malicious URL) and machine learning technique (to train the machine learning model for classification of malicious and benign URLs based on the attributes of malicious and benign URLs). Every technique is having its own pros and cons. In the blacklist technique, list of malicious URLs should be prepared through manual or automated system and also it requires frequent updating to detect the latest malicious URLs. Preparing such huge list of malicious URLs consumes more time and efforts, and this technique fails to detect the newly created malicious URLs. But it is fast enough to detect the malicious URL that is already in the blacklist [4]. The heuristic technique is more generalized approach than the blacklist. It uses the selected features to frame the rules for classifying malicious URLs from the benign. But preparing set of optimal number of features and assigning proper weightage or threshold value to the rules requires detailed investigation of the URLs [5]. Machine learning technique includes different algorithms for classification problem. Malicious URL detection is a binary classification method that employs a set of features derived from URLs (which includes both malicious and benign URLs) to train classification algorithms (models) to predict whether a newly produced URL is malicious or benign. Accuracy of the prediction or classification depends on the several factors such as choice of algorithms, selected features and amount of data used for training. Sometimes the prepared data to train the model may be overfitting or underfitting. Proper testing is essential for the model before deploying to real environment. Logistic regression, Support Vector Machine (SVM), k-Nearest Neighbor, decision tree, and Random Forest (RF) are examples of popular binary classification techniques.

The proposed method in this paper uses limited number of features with mixed type to train and test the models. Along with existing features, the newly introduced features in the dataset, makes the model to perform better than existing research works. The article provides brief information about the initial information of the study and existing research papers. The description of the proposed method includes the results of the experiment and the conclusion.

Background and Literature Review

Millions of web servers added in the internet to provide wider services to the clients which include billions of webpages. To locate and navigate each webpage uniquely requires an identifier called URL. The URL includes five components as shown in the below Fig. 1.

— Protocol — Determines the way of data communication between client and server.
— Domain Name — Uniquely identifiles the webserver in the internet. Using DNS server, the domain name is converted back to IP address of the web server in the internet. Domain name may have subdomain. The Top-Level Domain is always present in a domain name, and it may also include a second level domain.
— Port — Used to identify specific process in the webserver for getting response of the client request. But it's rarely visible in the URL.
— Path — Used to refer the specific resource of the webserver.
— Query String — Query string comes after the question mark (?) symbol in the URL which includes parameters and fragments.

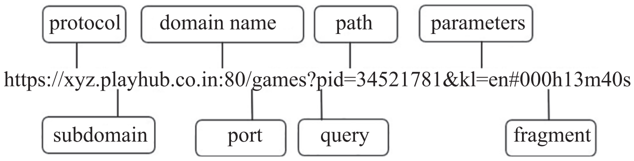
protocol
domain name
path
parameters
https://xyz.playhub.co.in:80/games?pid=34521781&kl=en#000h13m40s
subdomain
port
query
fragment

Fig. 1. Components of URL

- Parameters — Used to pass the values of some variable to a web page in the server to get dynamic web page. Usually, the parameters are in key and value format.
- Fragment — Used to refer the internal page reference.

Malicious and benign URLs appear to be identical in nature. To categorize them, different features from the URL, web page, and server information must be extracted. Table 1 shows how the features are divided into four categories. Extracting and analyzing lexical features is a faster and safer operation than other features. However, studying webpage contents and related properties is required to comprehend the dynamic nature of the URL. Despite the fact that a large number of research papers have been dedicated to URL categorization, the subject remains open and unsolved due to the changing nature of the assault and its signatures.

Apoorva Joshi et al. [4] highlight the role of machine and deep learning to find the mischievous URL, which is delivered through email. The proposed method used the static lexical features of URLs for URL classification. The dataset used for the experiment is derived from various sources such as openphish, alexa and fire eye and which include 60 % benign URL and 40 % malicious URLs. The Algorithm extracts 23 lexical features from the URL, and the test result reveals that it performs effectively with 95 % accuracy. Harshal Tupsamudre et al. [5] address the drawbacks of the BoW technique by classifying phishing URLs using word segmentation and n-grams, as well as traditional lexical aspects of URL and a phishy-list of popular terms. The experiment employed a dataset of 10,000 URLs gathered from various sources such as PhishTank and DMOZ. For testing, the Logistic Regression Algorithm is used, and the results reveal that it is more accurate than other approaches. Ozgur Koray et al. [11] proposed an anti-phishing system using Natural Language Processing (NLP) based features. Datasets were collected from PhishTank, Yandex Search API and Ebbu2017 Phishing Dataset and extracted the words (brand names, keywords, and random words) from the URL. Using those words, the required number of features is extracted for model to be trained and tested. Even though, the work outperforms the existing schemes, performance degradation occurs when large datasets are used for training. Patgiri et al. [12] proposed an efficient detection method for the prediction of malicious URL based on machine learning techniques. There seven machine learning algorithms were tested with dataset. For detecting phishing URLs, the RF Algorithm with NLP-based features only provides great results, with a 97.98 % accuracy ratio. Cho Do Xuan et al. [13] described a machine learning-based technique for detecting malicious URLs. For classification, the proposed approach comprises lexical, host, and content-based features. Bigdata technology is also used for improved speed of classification. The dataset was collected from different sources that include Phishtank, URLhaus and alexa. The experimental result shows that 96 % accuracy on RF over the SVM (91 % accuracy) classifier. Due to its speed of detection and safe browsing experience, Ferhat et al. [14] suggested a method for malicious URL identification using machine learning by using lexical and host-based features. The datasets for the experiment were obtained from the UCI Repository, and the features were extracted from a list of URLs. To select the suitable features for classification, PCA algorithm is used. The result shows RF model performed well (accuracy 98.6 %) over the gradient boosting model.

Butnaru et al. [15] proposed a machine learning-based lightweight solution for detecting malicious URLs. The dataset, which contains 305,737 benign URLs and 74,436 phishing URLs, was used to extract the limited lexical features. In the RF model, the result indicates 99.29 % accuracy.

Most of the researchers use lexical features to speed up feature generation and classification process. Some researchers included host and content based features along with lexical features. Word segmentation based features are also generated and utilized by some of the researchers. Although lexical features can be created quickly, they do not provide a robust detection system which considers the dynamics of a malicious URLs and webpages. The generation of the host and popularity based features requires access to third party servers that causes additional processing time. Content based features are generated by examining different components of the webpage and more importantly examining the components in the webpage, which can be utilized by the attackers. So visiting such pages and extracting the features becomes time consuming process and is not safe. Word segmentation based features require additional computation overhead. New forms of attacks are raised by using short URL and algorithmically generated URLs. Combining limited and essential features from different types of features will yield the better result.

Proposed System

Our proposed system considers most essential features which include lexical, host, content, popularity, and word segmentation. It also considers short URLs and algorithmically generated URLs. For our experimental purpose, required URLs are collected from UNB Database 2016, Phishtank and Kaggle which includes 10000 URLs

Table 1. URL features

Category	Description
Lexical features [5, 6]	Extracting features from URL such as count number of dots in URL, check the protocol (HTTP/HTTPS/FTP) of the URL, etc.
Host based features [5, 7, 8]	Extracting features from DNS/Web Server such as location of the webserver, date of registration, etc.
Content based features [9, 10]	Extracting features form the web page such as html features, JavaScript features, etc.
Reputation features [5]	Features related to rank of the webpage, social reputation, etc.

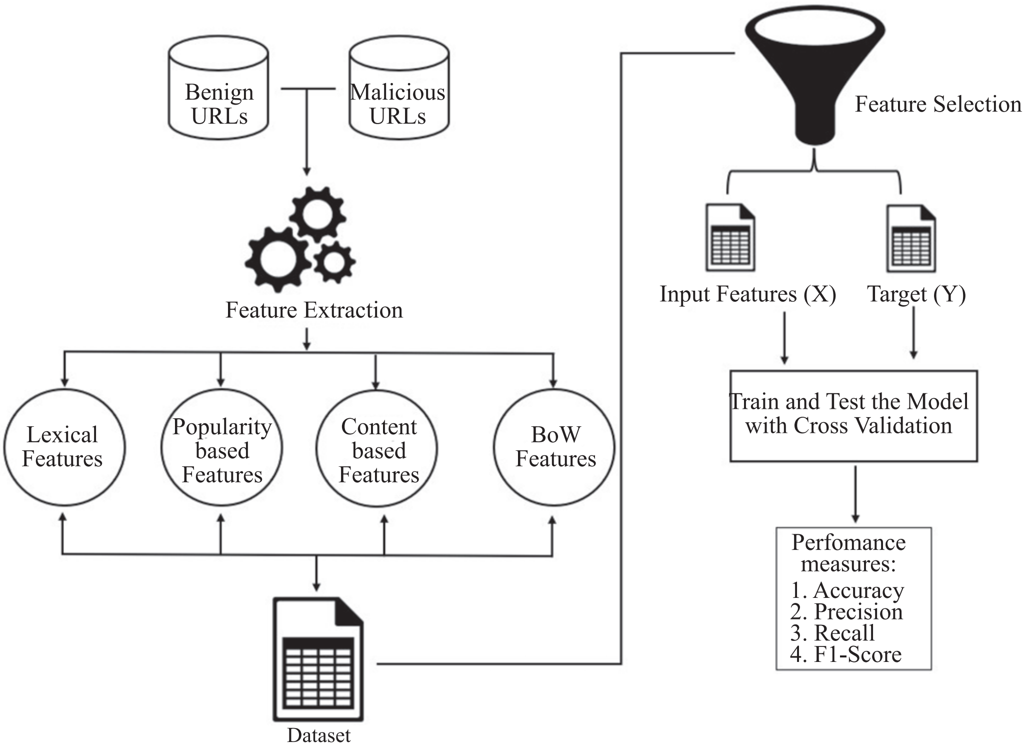

Fig. 2. Overview of the proposed system

(5000 Benign URLs and 5000 Malicious URLs). Fig. 2 depicts a high-level overview of the proposed system.

Feature Extraction

It's a process of identifying essential features from raw dataset. It is one of the preprocessing steps in data analysis. The output of the feature extraction process is the set of features and its values in tabular form such as CSV file. Table 2 below shows the features of our experiment. Table 2 includes lexical, popularity and content based features. The new features are marked by the * symbol. Feature extraction considers the short URLs, if the short URL is present, then it will be converted to original URL for further feature extraction. It also considers the algorithmically generated URL by computing the entropy of the URL. BoW (Bag of Words) includes 150 block listed words which help to check if the block listed words are present in the URL.

Feature Selection

Feature selection is required to remove unnecessary features from a dataset, lowering computational complexity and improving model performance [16]. Feature selection can be done through multiple ways such as chi-square method, correlation coefficient method, recursive feature elimination method, forward selection method, backward selection method, and lasso regularization. Among these, correction coefficient is simple method to understand the relationship between two features. Highly correlated features are removed from the dataset. The result of the feature selection includes only 26 features (feature no 1–7, 10–17, 19, 21–24, 26, 30–34) out of 34 features.

Experiment Results

The experimental configuration consists of a Windows 10 operating system, an Intel i5 processor running at 3.2 GHz, and 8 GB of RAM. Jupyter Notebook with sklearn package is used for programming. The accuracy is calculated using Table 3 and equation:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

The percentage of correct decisions among all testing samples is known as accuracy.

Precision, recall, and F1-score are three further performance metrics tested with the proposed system, employing formulas

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F1} - \text{Score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

For the comparison of result accuracy, SVM and RF classifiers are selected and compared with the results of the methods proposed Cho Do [11]. The results of experiment are presented in the Table 4 and Fig. 3. The result shows that proposed method performs better than existing method.

Table 2. Extracted features

No	Feature type	Feature name	Description
1	Lexical	domNum	Domain name contains an IP address
2	Lexical	presport	Domain name contains a port number
3	Lexical	entroUrl	Entropy of the URL
4	Lexical	entroDom*	Entropy of the domain name
5	Lexical	shortUrlPres*	Presence of the short URL
6	Lexical	faviconUrl	Presence of the favicon in URL
7	Lexical	lenUrl	Length of the URL
8	Lexical	lenPath	Path length in the URL
9	Lexical	lenParam*	Parameter's length in the URL
10	Lexical	lenQuery	Query's length in the URL
11	Lexical	lenFrag*	Length of the fragment in the URL
12	Lexical	lenDom	Domain's length in the URL
13	Lexical	numberCountDomain	Count the numbers in the domain
14	Lexical	dotCountDom	Count the dots in the domain
15	Lexical	atCountUrl	Count the at symbol in the URL
16	Lexical	equalCountUrl	Count the equal symbol in the URL
17	Lexical	undscrCountUrl	Count the underscore symbol in the URL
18	Lexical	slashCountUrl	Count the slash symbol in the URL
19	Lexical	hashCountUrl	Count the hash symbol in the URL
20	Lexical	andCountUrl	Count the "and" symbol in the URL
21	Lexical	questionCountUrl	Count the question symbol in the URL
22	Lexical	hyphenCountUrl	Count the hyphen symbol in the URL
23	Lexical	schemeCountUrl	Count the protocols in the URL
24	BoW	boWCountUrl	Count the presence of block listed words in the URL
25	Lexical	countAlphabetUrl	Count the alphabets in the URL
26	Lexical	countNumberUrl	Count the numbers in the URL
27	Lexical	countSpecUrl	Count the special characters in the URL
28	Lexical	ratioUrlDomLen*	Ratio between URL length & domain length in the URL
29	Lexical	rationAlphaNumUrl*	Ratio between alphabets & numbers in the URL
30	Lexical	ratioAlphaSplUrl*	Ratio between alphabets & special characters in the URL
31	Lexical	ratioNumSplUrl*	Ratio between numbers & special characters in the URL
32	Popularity	rankUrl	Global rank of the URL
33	Content	hrefCountContent	Count the href in the webpage
34	Content	iframeCountContent	Count the iframe in the webpage

Table 3. Confusion matrix

URLs	Classified	
	Malicious	Benign
Malicious	True Positive (TP)	False Negative (FN)
benign	False Positive (FP)	True Negative (TN)

Table 4. Result of classifiers

Methods	Dataset	Classifier	Accuracy, %	Precision, %	Recall, %	Execution time in seconds
Cho Do [11] method	10,000 URLs	SVM (10 Iteration)	93.35	94.84	92.71	3.12
		RF (10 Trees)	99.10	98.43	97.45	2.79
Proposed method		SVM (10 Iteration)	94.10	95	94.50	3.13
		RF (10 Trees)	99	99.50	99.50	0.17

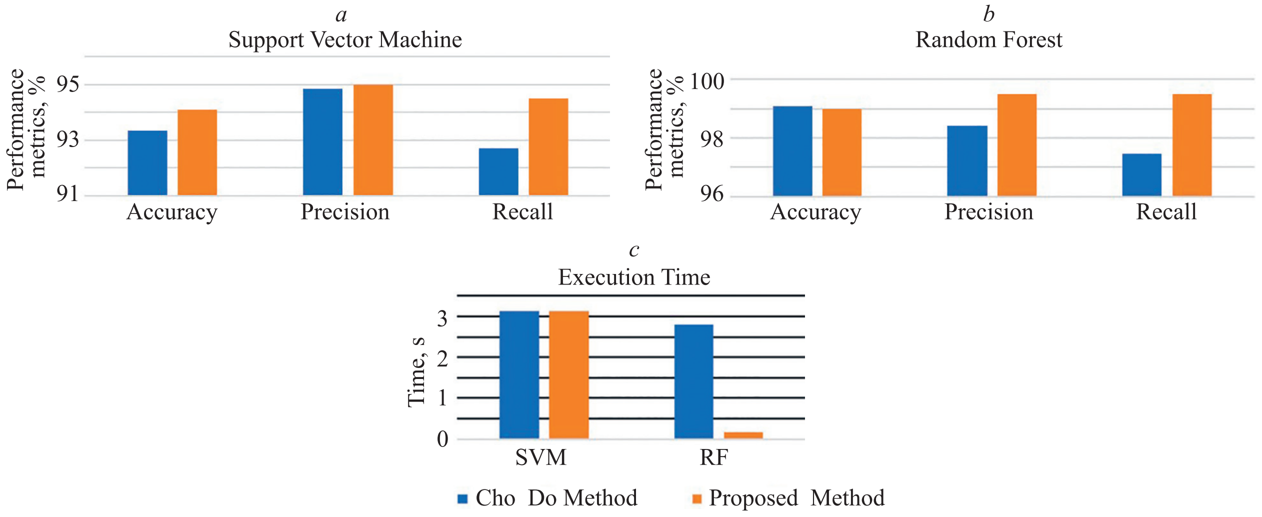

Fig. 3. Performance comparison: SVM (*a*), RF (*b*), execution time comparison (*c*)

Conclusion

The research work extract includes hybrid features of Uniform Resource Locator (URL) from 10,000 URLs set that are retrieved from different sources. Support Vector Machine and Random Forest results reveal that the suggested method outperforms the existing method. The research work can be extended by including URL and web page content analysis by using word segmentation and also adopting deep learning methods.

References

1. Warburton D. 2020 *Phishing and Fraud Report.* Available at: https://www.f5.com/labs/articles/threat-intelligence/2020-phishing-and-fraud-report (accessed: 11.11.2020).
2. Saleem Raja A., Vinodini R., Kavitha A. Lexical features based malicious URL detection using machine learning techniques. *Materials Today: Proceedings*, 2021, vol. 47, part 1, pp. 163–166. https://doi.org/10.1016/j.matpr.2021.04.041
3. Pradeepa G., Devi R. Review of malicious URL detection using machine learning. *Advances in Intelligent Systems and Computing*, 2021, vol. 1397, pp. 97–105. https://doi.org/10.1007/978-981-16-5301-8_7
4. Joshi A., Lloyd L., Westin P., Seethapathy S. Using lexical features for malicious URL detection — a machine learning approach. *ArXiv*, 2019, arXiv:1910.06277.
5. Tupsamudre H., Singh A.K., Lodha S. Everything is in the name — a URL based approach for phishing detection. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2019, vol. 11527, pp. 231–248. https://doi.org/10.1007/978-3-030-20951-3_21
6. Sahoo D., Liu C., Hoi S.C.H. Malicious URL Detection using Machine Learning: A Survey. *arXiv, 2017*, arXiv:1701.07179.
7. Ma J., Saul L.K., Savage S., Voelker G.M. Identifying suspicious URLs: an application of large-scale online learning. *Proc. of the 26th International Conference on Machine Learning (ICML)*, 2009, pp. 681–688. https://doi.org/10.1145/1553374.1553462
8. Kevin McGrath D., Gupta M. Behind phishing: An examination of phisher modi operandi. Proc. of the *1st USENIX Workshop on Large-*

Литература

1. Warburton D. 2020 Phishing and Fraud Report [Электронный ресурс]. URL: https://www.f5.com/labs/articles/threat-intelligence/2020-phishing-and-fraud-report (дата обращения: 11.11.2020).
2. Saleem Raja A., Vinodini R., Kavitha A. Lexical features based malicious URL detection using machine learning techniques // Materials Today: Proceedings. 2021. V. 47. Part 1. P. 163–166. https://doi.org/10.1016/j.matpr.2021.04.041
3. Pradeepa G., Devi R. Review of malicious URL detection using machine learning // Advances in Intelligent Systems and Computing. 2021. V. 1397. P. 97–105. https://doi.org/10.1007/978-981-16-5301-8_7
4. Joshi A., Lloyd L., Westin P., Seethapathy S. Using lexical features for malicious URL detection - a machine learning approach // arXiv. 2019. arXiv:1910.06277.
5. Tupsamudre H., Singh A.K., Lodha S. Everything is in the name — a URL based approach for phishing detection // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2019. V. 11527. P. 231–248. https://doi.org/10.1007/978-3-030-20951-3_21
6. Sahoo D., Liu C., Hoi S.C.H. Malicious URL Detection using Machine Learning: A Survey // arXiv. 2017. arXiv:1701.07179.
7. Ma J., Saul L.K., Savage S., Voelker G.M. Identifying suspicious URLs: an application of large-scale online learning // Proc. of the 26th International Conference on Machine Learning (ICML). 2009. P. 681–688. https://doi.org/10.1145/1553374.1553462

Scale Exploits and Emergent Threats: Botnets, Spyware, Worms, and More (LEET), 2008.

9. Hou Y.-T., Chang Y., Chen T., Laih C.-S., Chen C.-M. Malicious web content detection by machine learning. *Expert Systems with Applications*, 2010, vol. 37, no. 1, pp. 55–60. https://doi.org/10.1016/j.eswa.2009.05.023
10. Fu A.Y., Liu W., Deng X. Detecting phishing web pages with visual similarity assessment based on earth mover's distance (EMD). *IEEE Transactions on Dependable and Secure Computing*, 2006, vol. 3, no. 4, pp. 301–311. https://doi.org/10.1109/TDSC.2006.50
11. Sahingoz O.K., Buber E., Demir O., Diri B. Machine learning based phishing detection from URLs. *Expert Systems with Applications*, 2019, vol. 117, pp. 345–357. https://doi.org/10.1016/j.eswa.2018.09.029
12. Patgiri R., Katari H., Kumar R., Sharma D. Empirical study on malicious URL detection using machine learning. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2019, vol. 11319, pp. 380–388. https://doi.org/10.1007/978-3-030-05366-6_31
13. Xuan C.D., Nguyen H.D., Tisenko V.N. Malicious URL detection based on machine learning. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 2020, vol. 11, no. 1. http://doi.org/10.14569/IJACSA.2020.0110119
14. Catak F.O., Sahinbas K., Dörtkardeş V. Malicious URL detection using machine learning. *Artificial Intelligence Paradigms for Smart Cyber-Physical Systems*, 2021, pp. 21. https://doi.org/10.4018/978-1-7998-5101-1.ch008
15. Butnaru A., Mylonas A., Pitropakis N. Towards lightweight URL-based phishing detection. *Future Internet*, 2021, vol. 13, no. 6, pp. 154. https://doi.org/10.3390/fi13060154
16. Browniee J. *How to choose a feature selection method for machine learning*. Available at: https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/ (accessed: 20.08.2020).

8. Kevin McGrath D., Gupta M. Behind phishing: An examination of phisher modi operandi // Proc. of the 1st USENIX Workshop on Large-Scale Exploits and Emergent Threats: Botnets, Spyware, Worms, and More (LEET). 2008.
9. Hou Y.-T., Chang Y., Chen T., Laih C.-S., Chen C.-M. Malicious web content detection by machine learning // Expert Systems with Applications. 2010. V. 37. N 1. P. 55–60. https://doi.org/10.1016/j.eswa.2009.05.023
10. Fu A.Y., Liu W., Deng X. Detecting phishing web pages with visual similarity assessment based on earth mover's distance (EMD) // IEEE Transactions on Dependable and Secure Computing. 2006. V. 3. N 4. P. 301–311. https://doi.org/10.1109/TDSC.2006.50
11. Sahingoz O.K., Buber E., Demir O., Diri B. Machine learning based phishing detection from URLs // Expert Systems with Applications. 2019. V. 117. P. 345–357. https://doi.org/10.1016/j.eswa.2018.09.029
12. Patgiri R., Katari H., Kumar R., Sharma D. Empirical study on malicious URL detection using machine learning // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2019. V. 11319. P. 380–388. https://doi.org/10.1007/978-3-030-05366-6_31
13. Xuan C.D., Nguyen H.D., Tisenko V.N. Malicious URL detection based on machine learning // International Journal of Advanced Computer Science and Applications (IJACSA). 2020. V. 11. N 1. http://doi.org/10.14569/IJACSA.2020.0110119
14. Catak F.O., Sahinbas K., Dörtkardeş V. Malicious URL detection using machine learning // Artificial Intelligence Paradigms for Smart Cyber-Physical Systems. 2021. P. 21. https://doi.org/10.4018/978-1-7998-5101-1.ch008
15. Butnaru A., Mylonas A., Pitropakis N. Towards lightweight URL-based phishing detection // Future Internet. 2021. V. 13. N 6. P. 154. https://doi.org/10.3390/fi13060154
16. Browniee J. How to choose a feature selection method for machine learning [Электронный ресурс]. URL: https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/ (дата обращения: 20.08.2020).

Authors

Ganesan Pradeepa — Research Scholar, Vels Institute of Science, Technology and Advanced Studies, Pallavaram, Chennai, 600117, India, https://orcid.org/0000-0002-5920-066X, pradeepa25.ganesan@gmail.com

Radhakrishnan Devi — Associate Professor, Vels Institute of Science, Technology and Advanced Studies, Pallavaram, Chennai, 600117, India, SC 57195412460, https://orcid.org/0000-0002-8951-2242, devi.scs@velsuniv.ac.in

Received 09.02.2022
Approved after reviewing 22.02.2022
Accepted 17.03.2022

Авторы

Прадипа Ганесан — исследователь, Институт науки, технологий и перспективных исследований Велса, Паллаварам, Ченнаи, 600117, Индия, https://orcid.org/0000-0002-5920-066X, pradeepa25.ganesan@gmail.com

Деви Радхакришнан — доцент, Институт науки, технологий и перспективных исследований Велса, Паллаварам, Ченнаи, 600117, Индия, SC 57195412460, https://orcid.org/0000-0002-8951-2242, devi.scs@velsuniv.ac.in

Статья поступила в редакцию 09.02.2022
Одобрена после рецензирования 22.02.2022
Принята к печати 17.03.2022