Malicious URL Classification Using Machine Learning

Trinanjan Daw[1], Pourik Saha[2], Mainak Sen[3](✉), Khokan Mondal[3], and Amlan Chakrabarti[1]

[1] A. K. Choudhury School of Information Technology, University of Calcutta, Kolkata, India
acakcs@caluniv.ac.in

[2] Department of Computer Science, Benaras Hindu University, Varanasi, India

[3] Computer Science & Engineering Department, Techno India University, Kolkata, West Bengal, India
mainaksen.1988@gmail.com, khokan20@gmail.com

Abstract. In this era of emerging web technologies, people are dealing with online resources at all times. Web phishing is a social engineering attack where people are tricked by cyber attackers and reveal important credentials. People type a search query and based on the query results, users generally enter into top few websites. Cyber attackers deploy different mechanisms like keyword stuffing, content cloaking to modify the rank of a webpage. In this paper, we have checked whether a website is content cloaked or not and then further classified the URL whether it is phishing or not. We have used SVM and K-NN classifier on a dataset of 11,72,598 URLs and achieved 90% accuracy score.

Keywords: Content Cloaking · ssdeep · DNStwist · Malicious URL · Machine Learning

1 Introduction

In this 21st century, the infrastructure of global organizations are dependent on the web application. Web sites are framed to fool normal people in order to earn trustworthiness. Web phishing is a mechanism by which people gets fooled when they click on a URL(Uniform Resource Locator) that is sent to them through SMS, e-mail etc. The phrase **ph** [4]comes form phone phreaking. Anti Phishing Working Group(APWG) [1] reports total of 1,025,968 attacks in first quarter of 2022, financial services being targeted most.

[7] states that spreading malicious URL is ranked first amongst different types of cyber attacks. Phishing attacks and attacking techniques have been evolving over the last few years. Phishers use many obfuscation techniques to convince normal people about the legitimacy of a website. Attackers use old registered domain [2], insert top brand name in different parts of a URL to earn the believe of people and fool them.

S. Bhattacharyya et al. (Eds.): Human 2023, STHC, pp. 147–156, 2023.
https://doi.org/10.1007/978-981-99-3478-2_14

People uses search engine to query according to their interest and when search engine results pages(SERP) are shown, user clicks on one of the returned pages. A page's content and link data determine how well a search engine ranks it.

Content cloaking [6] is a mechanism by which different content is shown to user and search engine crawlers based on certain keywords. Content of a website influences the rank of that website and with a high rank web page similar to a top brand, people easily gets trapped. A content provider tries to obscure (cloak) the genuine content from the search engine's view by offering multiple copies of a web page to browsers and search engines. Cyber-criminals practise of showing different content to a search engine crawler than to a browser or user in order to encourage phishing.

Domain squatting is the act of purchasing a domain to block someone else from registering it, in order to get profit while reselling it, or for selling ads. These types of domains are called squatting domains. Attackers try to create a fake site that impersonates top brands and these domains are known as combo-squatting or typo-squatting. In, combo-squatting, people concatenate target domains with other characters or word. For example, www.face-book.com instead of www.facebook.com is being issued in public interest. Typo-Squatting is the art of writing the domain names with a typo error. For example, someone may type www.faecbook.com instead of www.facebook.com and these type of typo-squatting domains are generated from DNSTwist.

Major contribution of this paper are as follows: 1. We checked whether a given website(URL) is cloaked or not. 2. Based on the result(not cloaked) obtained from step 1, we have further classified a URL is either malicious or benign using machine learning based classifiers like SVM and K-NN.

The rest of the paper is organized as follows. In Sect. 2, literature survey is presented. Our proposed work is stated in Sect. 3 along with the description of the features used. Experiments and results are demonstrated in Sect. 4 and Sect. 5 deals with the conclusion and future scope of this presented work.

2 Survey

Tian et al. [8]revealed that 1175 URLs were mocking popular brands out of 657663 squatting domains. They used keywords, login forms, and other methods to identify malicious activities.

In their study of social engineering attacks delivered by malicious advertising, Vadrevu and Perdisci [9] discovered 11341 (16.1%) publisher sites containing malicious advertisements.

K. Ramya et al. [5] proposed the preventive measures against XSS attack. The way implemented is to keep a track on JS Event which could able to distinguish the malicious URL and benign URL. Authors used algorithms like SVC, KNeighborsRegressor, Random Forest Regressor, Gradient Boosting Regressor. Amongst these algorithms, Random Forest achieves the highest accuracy of 95.98% followed by Gradient Boosting with 95.8% and SVC achieved least accuracy of SVC 67%.

Chunlin Liu [3] proposed statistical overview of selecting best machine learning models. For constructing the models WEKA library was used. Some specific models were used like Random Forest, J48, Logistic Regression, LibSVm, Multilayer Perceptron and Naive Bayes.

Gold Wejinya [10] also classified URL phishing or benign using three classifiers namely SVM, Naive Bayes and logistic regression and compared in terms of the accuracy of these three classifiers. Total 11055 URLs were used consisting of both Malicious and benign types with 15 features. Author received 100% accuracy for Naive Bayes, 98% for Support Vector Machine(SVM) and 96% for logistic regression.

CANTINA [14] approach operates on the textual of website. TF-IDF is applied on the textual content material to extract excessive TF-IDF rating phrases and fed to the quest engine to identify as a phishing site or not. As the performance of this algorithm relies on the text data of the any webpage, it fails to deal with images and dependency on third party services needs extra time which is also a drawback.

CANTINA+, which integrates eight elements including HTML DOM features, third-party services, and search engine-related, was proposed by Xiang et al [11]. The authors have used six learning algorithms and bayesian network algorithm performed the best in comparison with other algorithms.

Zhang et al. [13] proposed a version together with URL and website content to detect chinese phishing e-commercial enterprise websites. Authors integrated fifteen features from domain for detecting phishing assaults in Chinese e-commercial enterprise websites. Out of four machine learning algorithms, Sequential minimal optimization (SMO) algorithm gave best accuracy of 95.83% but this approach could no longer work efficiently with non-Chinese websites.

3 Proposed Work

The architecture of the model is displayed in two phases. First, we have checked whether a website is cloaked or not and extracting features from URL as given in Table 1 for classification. It works on API where user given input content is checked to detect the cloaking behaviour of websites and next the classification based on the extracted feature takes place on the URL. Cloaking content is checked by ssdeep and then non-cloaked website is passed through URL feature extractor module. Figure 1 depicts the overall flow diagram of the proposed method.

DNSTwist is a domain name permutation engine. It starts with a specific domain name and generates a list of potential registered phishing/malware domains. DNSTwist passes the given seed domain to DomainFuzz and generates many random permutations of domain similar to seed domain. DNSTwist checks for content similarity using fuzzy hashes (Context Triggered Piecewise Hashes or CTPH). Fuzzy hashing, is a mechanism that compares two inputs (HTML code/compare the fuzzy hash of a squatting domain against the original domain) and determines a fundamental level of similarity.

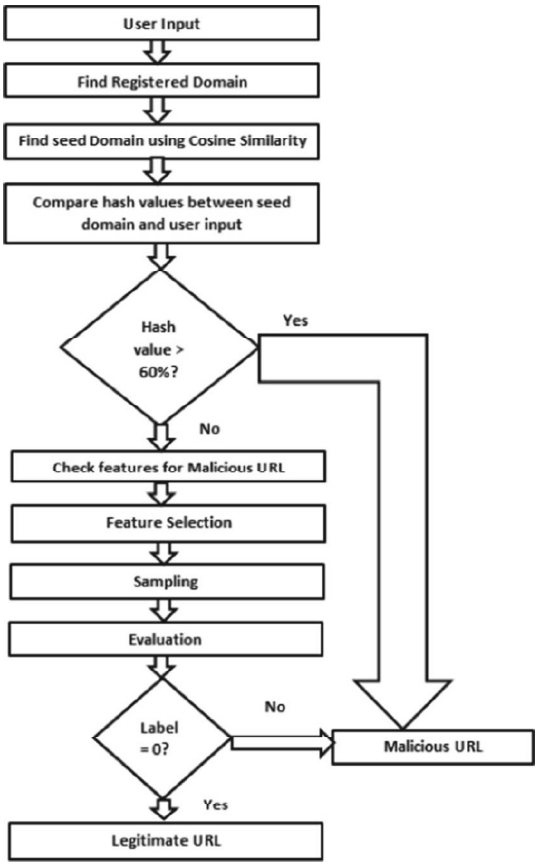


Fig. 1. Proposed System Architecture

Table 1. List of features Used for classification.

SL No.	Feature Description
1	IP Address
2	Count No. of Dots
3	Having Special @ symbol
4	Length of the URL
5	Depth of the URL
6	Position of the Redirection
7	Presence of http Domain in the URL
8	Tiny URL
9	Prefix and Suffix of the URL
10	Presence of the IFrame Tag
11	Presence of the Mouse Over event
12	Presence of onLoad Event
13	Presence of Right Click Event
14	Checking the History of the Website

3.1 List of Features Used for Classification

1. Identification of Ip(Internet Protocol)Address
   If URL has an IP address, then this feature is set to 1 else 0. As most legitimate website doesn't give the IP address to download the webpage, this behaviour detected in a URL identifies a suspicious URL.
2. Number of Dots
   The number of dots in a URL is counted. Phisher tries to hide the actual URL by adding more dots after the original domain to make it trustworthy.
3. Special Symbol in URL
   This feature also tries to hide the actual URL to make it phishy similar to existing top brand. For example, seeing the URL: http://www.login.flipkart.in/@www.malicious.com/abs/def/index.html, people might click on the link and might land up at www.malicious.com.
4. Length of URL
   Counting the number of Characters in the URL. Phishers tries to make the URL lengthy to hide the actual URL similar to the dot feature. Example: trinanjan.daw.RKMVCC.UG.Computer.Science.com. If the length of a URL is lagger than 54 characters, then we have set it to 1 to indicate phishing else 0 to indicate legitimate.
5. Depth of URL
   A numerical feature indicates the number of sub pages in a given url based on the '/' as delimiter. Example: sonu.com/sem/DS/oi/pen/
6. Presence of Redirection
   This feature checks the presence of "//" in the URL which means the user will be redirected to another website and the most important thing is finding the location of the "//" in a URL. If the URL starts with "HTTP", that means the "//" should appear in the sixth position and if it starts with "HTTPS" then the "//" should appear in seventh position. If it appears at location 6th or 7th then we set this to 0 else to 1 which indicates phishing.
7. Presence of https in the Domain
   Presence of http/https, in the domain indicates 1 or phishing URL else legitimate or 0. Example abcsdf.http.india.com.
8. Tiny URL
   URL shortening is a mechanism in which a URL can be made to appear smaller but leads to the desired webpage. If the URL is using Shortening Services, the value assigned to this feature is 1 (phishing) or else 0 (legitimate).
9. Prefix and suffix of the Domain
   Presence of any (-) character determines the Phishing URL(1) or else legitimate URL(0) like Example: cal-univ.ac.in. Phishers use - symbol in the domain part to fool users.
   Content Based Features
   Following are the features we have used for the classification from the content of a website.
10. Presence of IFrame
    It's a HTML tag which helps in embedding another web page to the current one. Attackers use frameborder attribute to portray the webpage. If the

response in iframe is empty, then this is set to 1 to indicate phishing else 0 to indicate legitimate.

11. Presence of onmouseover event
    This event helps to check the status bar URL. Phishers may replace the actual URL with Fake URL. Presence of the events tends to act as malicious.
12. Presence of onLoad event
    While downloading some software or a folder the website seeks our permission using a prompt. If such activities are detected then it recognized as malicious website(1) else benign website(0).
13. Presence of rightClick event
    Normal user can download the web-content by using the right clicking on it. But phishers hide their code from downloading their code. If the response is null then it is assumed to be a malicious URL/website.
14. Website forwarding
    This action is checked for how many times the website has been forwarded. If it exceeds the maximum limit(2) then recognised as malicious else legitimate.

3.2 Machine Learning Algorithms

3.2.1 Support Vector Machine

It is a supervised machine-learning model used for classification purposes. The goal of the model is to create the best fit line called hyperplane to segregate the classes so that we could easily put data into correct category. SVM chooses the two extremes so called support vectors to create the hyperplane. This decision boundary takes the elements extremes of the element or the element closest to the boundary to make the classification. There are various forms of SVM like Linear SVM, Kernel SVM etc. We have used Linear SVM specifically as we deal with binary value 0 for Legitimate and 1 for Phishing.

A support vector machine tries to minimise the distance between the distance between the two classes by maximize hyperplane from each of the feature vector.

3.2.2 K-Nearest Neighbours

It is a supervised machine learning model used for classification. It assumes that the predefined data is loaded and based on the similarity it classifies the new data. This model is thus also called Lazy learner algorithm as it doesn't learns from previous data rather it stores the information and based on the similarity it classifies the data.

4 Experiment and Results

Given an URL for suspecting the behaviour of URL i.e. whether legitimate or malicious. Our Model does it in two phases, first is cloaking and second one is feature extraction. The first phase, finds whether it's a registered domain or not and then finds the seed domain similar to given URL using cosine function. Then compare the hash values between the seed domain and given URL. If the hash

value exceeds a certain limit then Malicious or else it enter the second phase for further verification. Here feature extraction takes place and accordingly, it is decided that the URL is legitimate or not.

4.1 Dataset Information

To successfully terminate our model we deal with 11,72,598 data containing 5,84,909 legitimate URL and 5,87,689 malicious URL [12]. The dataset is spllited into two parts, one for training(80%) and other for testing(20%) where legitimate URLs are labelled as 0 and malicious labelled as 1.

4.2 Performance Metrics

We have used accuracy, precision, recall and F1 -score as our performance metrics.

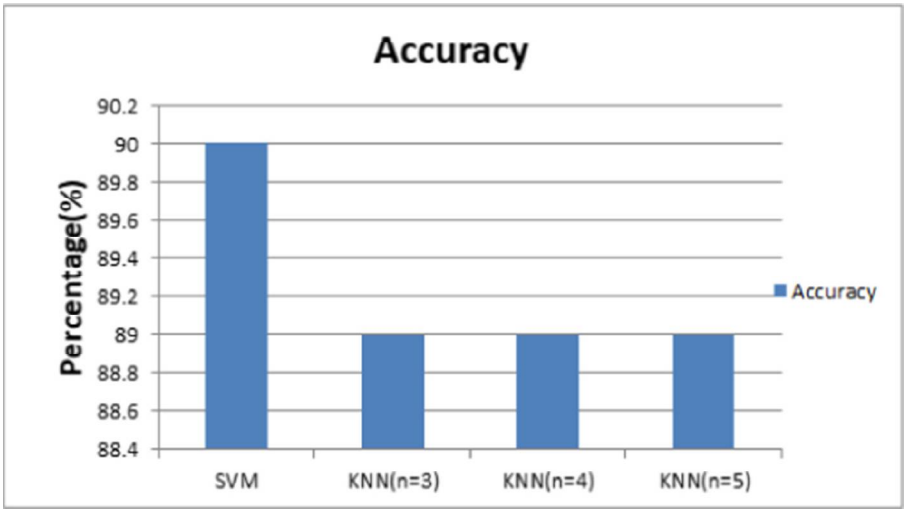
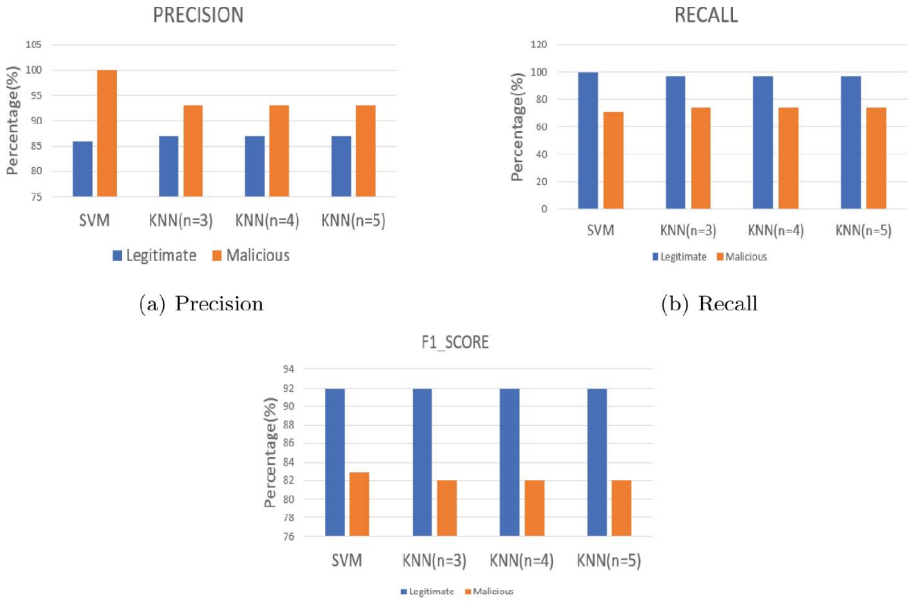


Fig. 2. Accuracy of Machine Learning Models.

The results conclude that the accuracy of SVM is the highest of 90% as it predicts by creating the best fit line to distinguish between the legitimate and malicious URL where as in KNN we have received 89% as shown in Fig. 2. Precision, recall and F1 -score are shown in Fig. 3.

We have checked our tool against different sample during run time. On giving the input 127.0.0.1 our model predicted the input as phishing as shown in Fig. 4 and Fig. 5 shows the predicted outcome of the model on giving the input sample caluniv.ac.in as legitimate.



(a) Precision

(b) Recall

(c) F1-score

Fig. 3. Performance Metrics.

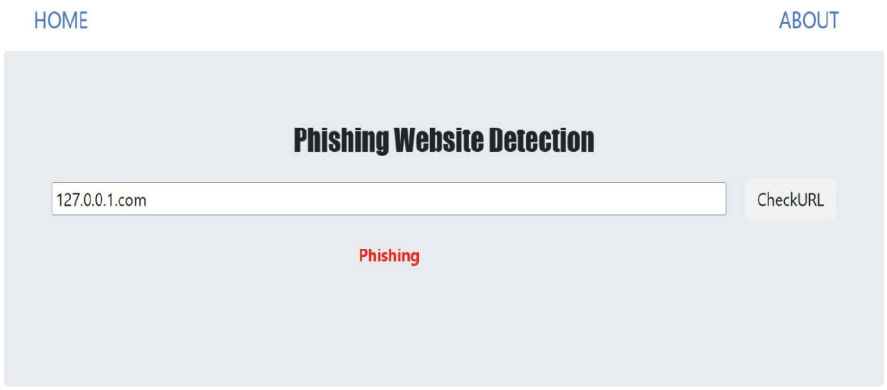


Fig. 4. Screenshot of Phishing Website Prediction.

Fig. 5. Screenshot of Legitimate Website Prediction.

5 Conclusion and Future Scope

Here, we have checked a URL/website whether they are cloaked or not which is needful to check against today's cyber attackers. Secondly, we have also used SVM and KNN for URL classification task and received better accuracy with support vector machine.

Further evaluations on additional data sets with additional features will be conducted and an API will be generated for public usage.

References

1. Anti-Phishing Working Group(APWG). https://apwg.org/trendsreports/ (Published on 7 June 2022)
2. Do Xuan, C., Dinh Nguyen, H., Nikolaevich Tisenko, V.: Malicious URL detection based on machine learning. Int. J. Adv. Comput. Sci. Appl. **11**(1), (2020)
3. Liu, C., Wang, L., Lang, B., Zhou, Y.: Finding effective classifier for malicious URL detection. In: Proceedings of the 2018 2nd International Conference on Management Engineering, Software Engineering and Service Sciences, pp. 240–244 (2018)
4. Ollmann, G.: The phishing guide understanding & preventing phishing attacks. NGS Software Insight Security Research (2004)
5. Ramya, K., Sharma, A., Mehta, K., Raj, V.: A comprehensive end-to-end framework for detection and prevention of cross site scripting attack
6. Samarasinghe, N., Mannan, M.: On cloaking behaviors of malicious websites. Comput. Secur. **101**, 102114 (2021)
7. Symantec. Internet security threat report (ISTR) (2019). https://www.symantec.com/content/dam/symantec/docs/reports/istr-24-2019-en.pdf (2019)
8. Tian, K., Jan, S.T.K., Hu, H., Yao, D., Wang, G.: Tracking down elite phishing domains in the wild: needle in a haystack. In: Proceedings of the Internet Measurement Conference, vol. 2018, pp. 429–442 (2018)
9. Vadrevu, P., Perdisci, R.: What you see is not what you get: discovering and tracking social engineering attack campaigns. In: Proceedings of the Internet Measurement Conference, pp. 308–321 (2019)

10. Wejinya, G., Bhatia, S.: Machine learning for malicious URL detection. In: Tuba, M., Akashe, S., Joshi, A. (eds.) ICT Systems and Sustainability. AISC, vol. 1270, pp. 463–472. Springer, Singapore (2021). https://doi.org/10.1007/978-981-15-8289-9_45
11. Xiang, G., Hong, J., Rose, C.P., Cranor, L.: Cantina+ a feature-rich machine learning framework for detecting phishing web sites. ACM Trans. Inf. Syst. Secur. (TISSEC) **14**(2), 1–28 (2011)
12. Yuan, H., Yang, Z., Chen, X., Li, Y., Liu, W.: URL2vec: URL modeling with character embeddings for fast and accurate phishing website detection. In: 2018 IEEE International Conference on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom), pp. 265–272 (2018)
13. Zhang, D., Yan, Z., Jiang, H., Kim, T.: A domain-feature enhanced classification model for the detection of Chinese phishing e-business websites. Inf. Manage. **51**(7), 845–853 (2014)
14. Zhang, Y., Hong, J.I., Cranor, L.F.: Cantina: a content-based approach to detecting phishing web sites. In: Proceedings of the 16th International Conference on World Wide Web, pp. 639–648 (2007)