Proactive Phishing Defense: A URL Classification System Using Machine Learning

Samer kadhim Jawad
Computer Engineering
Al-Iraqia University
Baghdad, Iraq
samer.k.jawad@aliraqia.edu.iq

Satea Hikmat Alnajjar
Network Engineering
Al-Iraqia University
Baghdad, Iraq
sateaahn@gmail.com

***Abstract*— Phishing attacks are the most common cyberattacks nowadays. Phishing attacks rely on social engineering concepts to trick victims into reaching the goals of malicious attackers. In addition, phishing attacks are the largest vector for various cyberattacks. However, URLs are a fulcrum for phishing attacks. The difficulty distinguishing between legitimate and phishing URLs is the reason for the increased success rates of these attacks. An integrated framework is proposed in this study to detect phishing attacks based on classifying URLs into phishing or legitimate URLs through machine learning models such as decision tree (DT) and random forest (RF), which have high power and prediction accuracy in binary classification tasks. The random forest model, using the cross validation (CV) technique, achieved an accuracy score of 98.2. This methodology is embedded in a web application with a graphical user interface to provide ease of handling and show alerts in real time and visually. This contributes to providing the field of cybersecurity with a highly accurate verification system to reduce users falling victim to these dangerous attacks.**

Keywords*— *Decision tree, Feature extraction, Phishing, Random Forest, URLs.

I. INTRODUCTION

Phishing attacks are attempts to deceive targets into divulging personal information, such as credit card numbers, usernames, and passwords. By taking advantage of human weaknesses like trust, urgency, and fear, attackers utilize social engineering (SE) techniques to pose as reputable companies like banks, well-known social media platforms, or government authorities [1].

Phishing attacks are one of the most common attacks used by attackers recently [2]. The ease of implementation and high success rates in achieving attackers' illicit goals are the main reasons for the widespread use and strong reliance on phishing attacks by attackers [3]. In addition, a phishing attack can be adopted as a stand-alone attack or can be a starting point for cyberattacks of another type [4]. However, tricking victims by impersonating official web page designs and launching phishing URLs to redirect victims to fraudulent pages are the two main methods that attackers rely on to create phishing attacks [5].

The difficulty in identifying phishing assaults poses the true risk. These harmful attacks can even affect experts and businesses with strong security protocols. In 2017, for example, a hacker successfully carried out a phishing attack against Google and Facebook, resulting in losses of up to $100 million [6].

On the other hand, researchers are making great efforts to provide methods to identify these malicious attacks. These methods include classifying URLs or checking the content of websites. In contrast to detection techniques that depend on content analysis, URL verification offers the advantage of requiring less time and resources to identify phishing attempts. Furthermore, provides real time verification responses [7].

Therefore, machine learning emerges as one of the most important concepts for detecting phishing attacks due to its ability to classify URLs into legitimate or phishing URLs through machine learning algorithms for binary classification tasks [8].

The study aims to present an integrated framework in the form of a client-side web application with a graphical user interface to detect phishing URL attacks in a proactive and real-time manner, relying on machine learning models random forest (RF), and decision tree (DT).

Additionally, highlights the features associated with URLs, the random forest model, and the bagging technique, while evaluating the performance of the models based on the cross validation (CV) technique, which explains the results achieved across multiple folds.

Finally, the study contributes to proposing a deterrence system against phishing attacks based on detecting malicious URLs through the actual application of a random forest ensemble learning classifier within a web application and achieves significant results through cross validation techniques to reduce overfitting and variance.

The remaining parts of the paper are organized as follows: The theoretical background related to phishing attacks, URLs, and machine learning models is addressed in Section II. Details of related work are given in Section III. However, the methodology applied to create the proposed framework is presented in Section IV. In the section V, the results are presented and discussed. Section VI concludes the paper and future work.

II. BACKGROUND

In this section of the study, the stages of a phishing attack are highlighted, as well as the basic components of URLs and machine learning models for classification tasks, such as random forest and decision tree.

A. Stages of implementing phishing attacks

Phishing attacks involve two main participants. The first is the attacker, who represents the source of the attack, and the victim, who represents the target. As mentioned earlier, phishing attacks are easy to carry out; All it takes is for the attacker to create a phishing website that bears a strong visual resemblance to a trusted, authentic website, a URL that points to the phishing website, and an email that looks authentic and includes the URL. In return, the victim will click on the URL in the email after reading it and being satisfied with its contents. This will take the victim to the phishing site, where they will choose to enter their personal information. Such as those related to bank accounts, login data, and the like, but the

party receiving this information will not be a legitimate party, but rather the attacker who will use this information according to his evil intentions [9]. Fig. 1 shows the stages of executing a phishing attack.

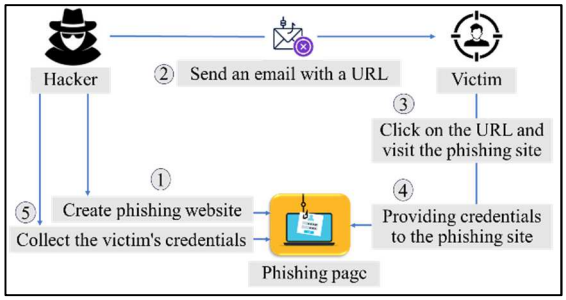

Fig. 1. Stages of a phishing attack

B. *Uniform Resource Locator (URL)*

Finding the location of resources on the Internet is the purpose of a URL. The three primary components of a URL are the hostname, path, and protocol. as depicted in Fig. 2. Nonetheless, in contemporary URL phishing assaults, malefactors exhibit their inventiveness by breaching the security regulations of safe HTTPS protocols via Heartbleed assaults and SSL truncation. Additionally, attackers can alter written hostname forms by depending on Typo-Squatting. Ultimately, the tactics of URL shortening and amplification are what give a phishing URL the appearance that the attacker desires.[10],[11].

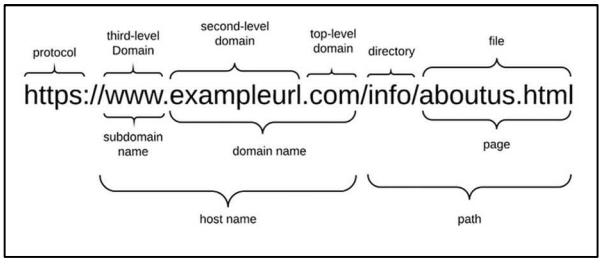

Fig. 2.URL Structure

However, to detect URL-based phishing attacks, four types of features can be extracted from URLs: address bar features, anomalous features, domain features, and HTML and JavaScript features[12].

C. *Machine learning models for URL classification*

In machine learning, classification refers to the process of predicting qualitative responses. Machine learning offers a wide range of models, including random forest (RF), decision tree(DT), and many others for binary classification tasks.

- *Decision tree*

One of the most popular machine learning algorithms, it has wide uses in classification and regression tasks. However, the decision tree algorithm has several contributions to classification tasks, especially in the binary classification of URLs (phishing or legitimate). The decision tree algorithm is simple and makes predictive decisions very similar to human decision making. The structure of the algorithm is similar to a tree, consisting of nodes representing features, and leaves representing the classification decision [13].

- *Random forests*

The Random Forest algorithm is one of the ensemble learning algorithms that rely on building several decision tree models as sub classifiers and training these sub classifiers through the bagging technique, where they are trained in a separate (parallel) way, collecting the final predictions, and choosing to predict the final result through a majority vote. This in turn helps to stabilize the model and the accuracy of the results, in addition to reducing variance and overfitting[14].

III. RELATED WORKS

N. Alam et al. [15] The researchers proposed a methodology to detect phishing attacks based on classifying URLs using decision tree classifiers and random forests, and from this study, they reached accuracy results of up to 97% through the random forest model.

A. Taha [16] The researcher proposed a model based on the bagging technique to build six heterogeneous classifiers and make predictions through soft voting. The model achieved maximum accuracy results of up to 95%.

K. R. Nataraj et al. [17] The researchers presented an integrated URL classification framework to detect phishing attacks by proposing ten machine learning classifiers. Through the gradient boosting classifier, they achieved accuracy results of up to 97.4%.

A. Karim et al. [18] The researchers relied on a hybrid model of three classifiers (LR+SVC+DT) to detect phishing attacks based on URLs, and through cross-validation technology, accuracy results of up to 98% were achieved.

K. Adane et al. [19] In their study, the researchers relied on three data sets and proposed three classifications for random forests, gradient boosting, cat boosting, and stacking technique. Through the Cat boost model, a Kaggle dataset containing 32 features, and 10-fold cross-validation, they achieved accuracy results of 97.36%.

IV. METHODOLOGY

The methodology followed in this study is based on creating an integrated phishing attack detection system based on the verification of URLs provided by end users by proposing a client-side web application with a graphical user interface. Fig. 3 shows the methodology of the proposed detection system.

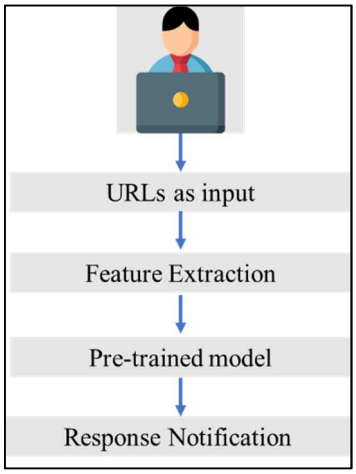

Fig. 3. Methodology

The proposed web application is created with the Flask framework, a web application development toolkit for Python, and includes a place for users to input URLs to be classified as phishing or authentic.

However, users provide URLs in the overall structure, which prediction algorithms cannot validate unless they are broken down into their parts, namely protocol, hostname, and path. Through the URL parse function, URLs are broken down into their three main components.

In addition, URLs are verified by checking a set of features. URLs have four categories of features: address bar features, anomalous features, HTML and JavaScript features, and domain features. Fig. 4 shows the URL feature categories and the features included in each category that are extracted from URLs by the Beautiful Soup and WHOIS libraries and third-party services.

Feature Categories	Features
Address bar Features	Using IP, Long URL, Short URL, Symbol@, Redirecting//, Prefix-Suffix, HTTPS, Domain registration, Favicon, Port
Abnormal Features	Request URL, Anchor URL, Script Tags, Server Handler, Info Email, Abnormal URL
HTML and JavaScript Features	Website Forwarding, Status Bar Cust, Disable Right Click, Using Popup Window, I frame Redirection
Domain Features	Age of Domain, DNS Recording, Website Traffic, Page Rank, Google Index, Links Pointing To Page , Stats Report

Fig. 4. URL features

On the other hand, machine learning classifiers random forest and decision tree are used to classify URLs provided by users after dividing them into their main components and extracting the necessary features from them to obtain an accurate binary classification (legitimate or phishing). Fig. 5 shows the flowchart of the training and testing process for the proposed models.

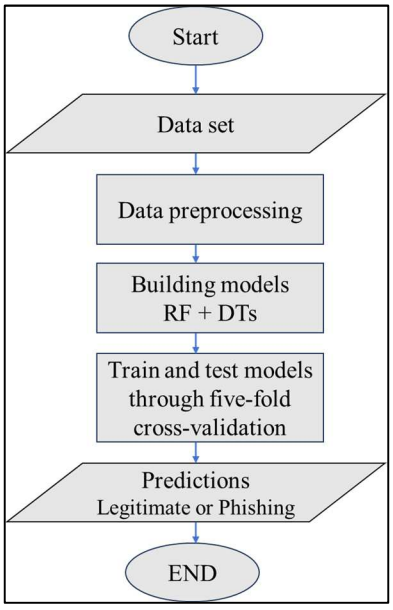

Fig. 5. Flowchart for training and testing the proposed models

Firstly, a dataset from the Kaggle repository is adopted, which includes 32 features with 11,000 instances labeled with -1 and 1 indicating phishing and legitimacy, which are well processed such that they have no missing values and do not require encoding.

However, unnecessary columns such as the index column were removed to avoid boring the models by training on unnecessary features and dividing the features into independent features (X) and dependent features (Y).

In addition, two classifiers have been proposed: decision tree and random forest classifiers, due to their robust and accurate predictions in binary classification tasks.

Finally, the proposed models are trained and tested based on the five-fold cross validation technique, selecting the model with the highest accuracy result, and saving this model in the pickle file in the Python environment to be called by the proposed detection system to verify the classification of URLs into legitimate addresses or phishing addresses.

In conclusion, final notifications are displayed to users regarding legitimate or phishing URLs provided in real time on the main graphical interface of the proposed web application, with a percentage indicating the reliability of the predictions, with the feature included of actual visits to legitimate URLs.

V. Results and Discussions

The proposed dataset for training machine learning models is taken from the Kaggle repository and is diverse in its features, covering most URL features and a fair number of instances included in them. Fig. 6 shows the features of the proposed dataset.

Index	UsingIP	LongURL	ShortURL
Symbol@	Redirecting//	PrefixSuffix-	SubDomains
HTTPS	DomainRegLen	Favicon	NonStdPort
HTTPSDomainURL	RequestURL	AnchorURL	LinksInScriptTags
ServerFormHandler	InfoEmail	AbnormalURL	WebsiteForwarding
StatusBarCust	DisableRightClick	UsingPopupWindow	IframeRedirection
AgeofDomain	DNSRecording	WebsiteTraffic	PageRank
GoogleIndex	LinksPointingToPage	StatsReport	class

Fig. 6. Features of the proposed data set

Proposed a random forest classifier and a decision tree for the key role they play in classification tasks. The comparison between the two proposed classifiers is to explore the results achieved by the decision tree classifier and the random forest classifier, which is considered an improvement to the decision tree model as it relies on collecting many decision tree classifiers and generating the final results through voting. Table 1 shows the results achieved by the random forest model and decision trees.

Table 1:Final results

Model	Accuracy	Precision	Recall	F1 Score
RF	98.2	97.9	98.8	98.4
DT	98.1	97.9	98.7	98.3

The achieved results were obtained based on the highest five-fold cross-validation results. Although the difference in results is small due to training and testing on a formatted and optimized dataset, the difference in results can be larger when using a different dataset.

However, the achieved results are significant and higher compared to existing works based on the same proposed dataset, indicating the robustness of the predictions of the random forest model in URL classification. Fig. 7 shows the accuracy results achieved by the random forest model in detail across five cross-validation folds.

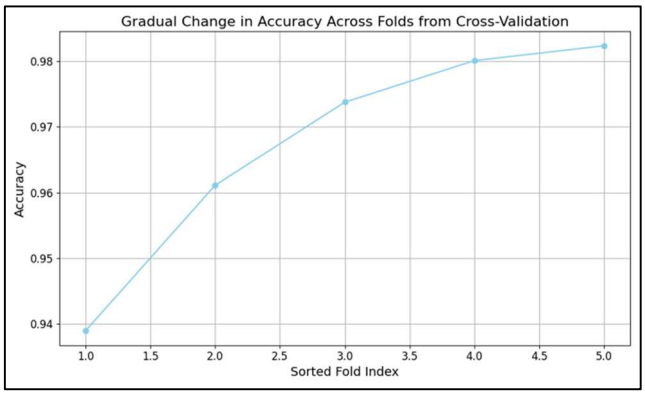

Fig. 7. RF accuracy by five-fold cross-validation

In addition, the results of the confusion matrix of the random forest model were adopted to calculate the final results of the proposed model, as shown in Fig. 8.

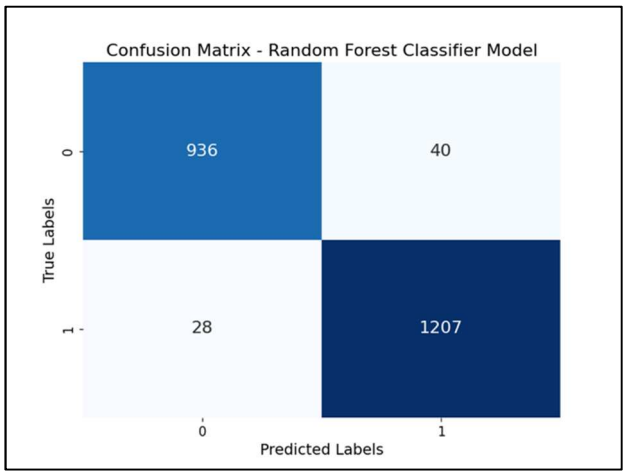

Fig. 8. RF confusion matrix

Where the accuracy results are obtained according to Equation 1, the Precision results are obtained according to Equation 2, in addition to the Recall results according to Equation 3, and finally the F1 score results according to Equation 4.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

$$\text{F1 score} = 2 * \frac{Precision*Recall}{Precision+Recall} \quad (4)$$

Finally, the random forest model is saved in a pickle file in the Python environment, which can then be called in the proposed detection system to classify URLs as legitimate or phishing. However, the proposed system is designed to be presented to users in the form of a web application with a user-friendly graphical interface and real-time urgent and immediate alerts. Fig. 9 shows the graphical interface of the proposed web application with an example to check the legitimacy of the Google website and the prediction reliability percentage.

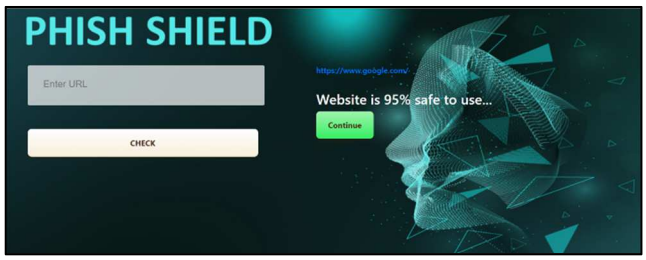

Fig. 9. The main interface of the proposed web application

VI. Conclusion and Future Work

Phishing attacks are one of the most widespread attacks and a real threat at various levels, whether individuals or organizations. URLs play a pivotal role in phishing attacks because they are the key to a phishing attack. They manipulate victims by exploiting users' inability to distinguish between legitimate and phishing URLs. Therefore, the web application is proposed in this study as an effective tool to help end users classify URLs into (legitimate or phishing) and a realistic shield against phishing attacks. The proposed web application relies on machine learning models for prediction through decision tree and random forest models to keep pace with advances in tactics used in modern phishing attacks. The results achieved through the random forest model are considered very good compared to the works currently presented, and this opens the door to other new works to search for other models that can achieve higher results.

Therefore, future work will focus on including other models, such as ensemble learning models based on boosting technology or attempting to suggest a model based on stacking technology to get more precise outcomes.

References

[1] M. A. Ivanov, B. V Kliuchnikova, I. V Chugunkov, and A. M. Plaksina, "Phishing Attacks and Protection Against Them," in *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*, 2021, pp. 425–428. doi: 10.1109/ElConRus51938.2021.9396693.

[2] C. Rajeswary and M. Thirumaran, "A Comprehensive Survey of Automated Website Phishing Detection Techniques: A Perspective of Artificial Intelligence and Human Behaviors," in *2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, 2023, pp. 420–427. doi: 10.1109/ICSCDS56580.2023.10104988.

[3] T. Egharevba, "Phishing Attack-A Challenge in Cybersecurity." 2022. doi: 10.13140/RG.2.2.32196.96640.

[4] B. Naqvi, K. Perova, A. Farooq, I. Makhdoom, S. Oyedeji, and J. Porras, "Mitigation strategies against the phishing attacks: A systematic literature review," *Comput. Secur.*, vol. 132, p. 103387, 2023, doi: https://doi.org/10.1016/j.cose.2023.103387.

[5] Y. Ding, N. Luktarhan, K. Li, and W. Slamu, “A keyword-based combination approach for detecting phishing webpages,” *Comput. Secur.*, vol. 84, pp. 256–275, 2019, doi: https://doi.org/10.1016/j.cose.2019.03.018.

[6] S. Albakry, K. Vaniea, and M. K. Wolters, “What is this URL’s Destination? Empirical Evaluation of Users’ URL Reading,” *Conference on Human Factors in Computing Systems - Proceedings*. 2020. doi: 10.1145/3313831.3376168.

[7] R. Zieni, L. Massari, and M. C. Calzarossa, “Phishing or not phishing? A survey on the detection of phishing websites,” *IEEE Access*, vol. 11, pp. 18499–18519, 2023.

[8] S. Abad, H. Gholamy, and M. Aslani, “Classification of Malicious URLs Using Machine Learning.,” *Sensors (Basel).*, vol. 23, no. 18, Sep. 2023, doi: 10.3390/s23187760.

[9] Q. A. Al-Haija and A. Al Badawi, “URL-based phishing websites detection via machine learning,” in *2021 International Conference on Data Analytics for Business and Industry (ICDABI)*, IEEE, 2021, pp. 644–649.

[10] L. Joon Sern and Y. Gui Peng David, “TypoSwype: An Imaging Approach to Detect Typo-Squatting,” in *2021 11th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*, 2021, pp. 1–5. doi: 10.1109/NTMS49979.2021.9432673.

[11] J. He, D. Han, and K.-C. Li, “On one-time cookies protocol based on one-time password,” *Soft Comput.*, vol. 24, no. 8, pp. 5657–5670, 2020, doi: 10.1007/s00500-019-04138-5.

[12] K. Mridha, J. Hasan, S. D, and A. Ghosh, “Phishing URL Classification Analysis Using ANN Algorithm,” in *2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON)*, 2021, pp. 1–7. doi: 10.1109/GUCON50781.2021.9573797.

[13] S. Alnemari and M. Alshammari, “Detecting Phishing Domains Using Machine Learning,” *Applied Sciences (Switzerland)*, vol. 13, no. 8. 2023. doi: 10.3390/app13084649.

[14] A. D. Purwanto, K. Wikantika, A. Deliar, and S. Darmawan, “Decision Tree and Random Forest Classification Algorithms for Mangrove Forest Mapping in Sembilang National Park, Indonesia,” *Remote Sens.*, vol. 15, no. 1, 2023, doi: 10.3390/rs15010016.

[15] M. N. Alam, D. Sarma, F. F. Lima, I. Saha, R.-E.-. Ulfath, and S. Hossain, “Phishing Attacks Detection using Machine Learning Approach,” in *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 2020, pp. 1173–1179. doi: 10.1109/ICSSIT48917.2020.9214225.

[16] A. Taha, “Intelligent ensemble learning approach for phishing website detection based on weighted soft voting,” *Mathematics*, vol. 9, no. 21. 2021. doi: 10.3390/math9212799.

[17] K. Nataraj, D. Yashaswini, R. Hema, N. Pawar, and S. Yashaswi, “Phishing Attack Detection Using Machine Learning,” 2023, pp. 355–370. doi: 10.1007/978-981-99-2058-7_33.

[18] A. Karim, M. Shahroz, K. Mustofa, S. B. Belhaouari, and S. R. K. Joga, “Phishing Detection System Through Hybrid Machine Learning Based on URL,” *IEEE Access*, vol. 11. pp. 36805–36822, 2023. doi: 10.1109/ACCESS.2023.3252366.

[19] K. Adane, B. Beyene, and M. Abebe, “Single and Hybrid-Ensemble Learning-Based Phishing Website Detection: Examining Impacts of Varied Nature Datasets and Informative Feature Selection Technique,” *Digit. Threat.*, vol. 4, no. 3, Oct. 2023, doi: 10.1145/3611392.