Detection of Phishing Websites Using Machine Learning

Madupu Venkata Vineeth[1], G.V. Ramana[2]

[1]*PG Student in Dept of CSE in Sree Vahini Institute of Science and Technology By-Pass Road, India*
[2]*Associate Professor in Dept of CSE in Sree Vahini Institute of Science and Technology By-Pass Road, India*

Abstract: Phishing websites have proven to be a major security concern. Several cyberattacks risk the confidentiality, integrity, and availability of company and consumer data, and phishing is the beginning point for many of them. Many researchers have spent decades creating unique approaches to automatically detect phishing websites. While cutting-edge solutions can deliver better results, they need a lot of manual feature engineering and aren't good at identifying new phishing attacks. As a result, finding strategies that can automatically detect phishing websites and quickly manage zero-day phishing attempts is an open challenge in this field. The web page in the URL which hosts that contains a wealth of data that can be used to determine the web server's maliciousness. Machine Learning is an effective method for detecting phishing. It also eliminates the disadvantages of the previous method. We conducted a thorough review of the literature and suggested a new method for detecting phishing websites using features extraction and a machine learning algorithm. The goal of this research is to use the dataset collected to train ML models and deep neural nets to anticipate phishing websites.

1. Introduction

Phishing has become the most serious problem, harming individuals, corporations, and even entire countries. The availability of multiple services such as online banking, entertainment, education, software downloading, and social networking has accelerated the Web's evolution in recent years. As a result, a massive amount of data is constantly downloaded and transferred to the Internet. Spoofed emails pretending to be from reputable businesses and agencies are used in social engineering techniques to direct consumers to fake websites that deceive users into giving financial information such as usernames and passwords. Technical tricks involve the installation of malicious software on computers to steal credentials directly, with systems frequently used to intercept users' online account usernames and passwords.

• Deceptive Phishing: This is the most frequent type of phishing assault, in which a Cybercriminal impersonates a well-known institution, domain, or organization to acquire sensitive personal information from the victim, such as login credentials, passwords, bank account information, credit card information, and so on. Because there is no personalization or customization for the people, this form of attack lacks sophistication. • Spear Phishing: Emails containing malicious URLs in this sort of phishing email contain a lot of personalization information about the potential victim. The recipient's name, company name, designation, friends, co-workers, and other social information may be included in the email. • Whale Phishing: To spear phish a "whale," here a top-level executive such as CEO, this sort of phishing targets corporate leaders such as CEOs and top-level management employees. • URL Phishing: To infect the target, the fraudster or cyber-criminal employs a URL link. People are sociable creatures who will eagerly click the link to accept friend invitations and may even be willing to disclose personal information such as email addresses. This is because the phishers are redirecting users to a false web server. Secure browser connections are also used by attackers to carry out their unlawful actions. Due to a lack of appropriate tools for combating phishing attacks, firms are unable to train their staff in this area, resulting in an increase in phishing attacks. Companies are educating their staff with mock

phishing assaults, updating all their systems with the latest security procedures, and encrypting important Information as broad countermeasures. Browsing without caution is one of the most common ways to become a victim of this phishing assault. The appearance of phishing websites is like that of authentic websites.

2. Literature Review

Many scholars have done some sort of analysis on the statistics of phishing URLs. Our technique incorporates key concepts from past research. We review past work in the detection of phishing sites using URL features, which inspired our current approach. Happy describe phishing as "one of the most dangerous ways for hackers to obtain users' accounts such as usernames, account numbers and passwords, without their awareness." Users are ignorant of this type of trap and will ultimately, they fall into Phishing scam. This could be due to a lack of a combination of financial aid and personal experience, as well as a lack of market awareness or brand trust. In this article, Mehmet et al. suggested a method for phishing detection based on URLs. To compare the results, the researchers utilized eight different algorithms to evaluate the URLs of three separate datasets using various sorts of machine learning methods and hierarchical architectures. The first method evaluates various features of the URL; the second method investigates the website's authenticity by determining where it is hosted and who operates it; and the third method investigates the website's graphic presence.We employ Machine Learning techniques and algorithms to analyse these many properties of URLs and websites. Garera et al. classify phishing URLs using logistic regression over hand-selected variables. The inclusion of red flag keywords in the URL, as well as features based on Google's Web page and Google's Page Rank quality recommendations, are among the features. Without access to the same URLs and features as our approach, it's difficult to conduct a direct comparison. In this research, Yong et al. created a novel approach for detecting phishing websites that focuses on detecting a URL which has been demonstrated to be an accurate and efficient way of detection. To offer you a better idea, our new capsule-based neural network is divided into several parallel components. One method involves removing shallow characteristics from URLs. The other two, on the other hand, construct accurate feature representations of URLs and use shallow features to evaluate URL legitimacy. The final output of our system is calculated by adding the outputs of all divisions. Extensive testing on a dataset collected from the Internet indicate that our system can compete with other cutting-edge detection methods while consuming a fair amount of time. For phishing detection, Vahid Shahrivari et al. used machine learning approaches. They used the logistic regression classification method, KNN, Adaboost algorithm, SVM, ANN and random forest. They found random forest algorithm provided good accuracy. Dr.G. Ravi Kumar used a variety of machine learning methods to detect phishing assaults. For improved results, they used NLP tools. They were able to achieve high accuracy using a Support Vector Machine and data that had been pre-processed using NLP approaches. Amani Alswailem et al. tried different machine learning model for phishing detection but was able to achieve more accuracy in random forest. Hossein et al. created the “Fresh-Phish” open-source framework. This system can be used to build machine-learning data for phishing websites. They used a smaller feature set and built the query in Python. They create a big, labelled dataset and test several machine-learning classifiers on it. Using machine-learning classifiers, this analysis yields very high accuracy. These studies look at how long it takes to train a model. X. Zhang suggested a phishing detection model based on mining the semantic characteristics of word embedding, semantic feature, and multi-scale statistical features in Chinese web pages to detect phishing performance successfully. To obtain statistical aspects of web pages, eleven features were retrieved and divided into five classes. To obtain statistical aspects of web pages, eleven features were retrieved and divided into five classes. To learn and evaluate the model, AdaBoost, Bagging, Random Forest, and SMO are utilized. The legitimate URLs dataset came from DirectIndustry online guides, and the phishing data came from China's Anti-Phishing Alliance. With novel methodologies, M. Aydin approaches a framework for extracting characteristics that is versatile and straightforward. Phish Tank provides data, and Google provides authentic URLs. C# programming and R programming were utilized to btain the text attributes. The dataset and third-party service providers yielded a total of 133 features. The feature selection approaches of CFS subset based and Consistency subset-based feature selection were employed and examined with the WEKA tool. The performance of the Nave Bayes and Sequential Minimal Optimization (SMO) algorithms was evaluated, and the author prefers SMO to NB for phishing detection.

3. Existing System

Anti-phishing strategies involve educating netizens and technical defines. In this paper, we mainly review the technical defines methodologies proposed in recent years. Identifying the phishing website is an efficient method in the whole process of deceiving user information Along with the development of machine learning techniques, various machine learning based methodologies have emerged for recognizing phishing websites to increase the performance of predictions. The primary purpose of this paper is to survey effective methods to prevent phishing attacks in a real-time environment.

4. Proposed System

The most frequent type of phishing assault, in which a cybercriminal impersonates a well-known institution, domain, or organization to acquire sensitive personal information from the victim, such as login credentials, passwords, bank account information, credit card information, and so on. Emails containing malicious URLs in this sort of phishing email contain a lot of personalization information about the potential victim. To spear phish a "whale," here a top-level executive such as CEO, this sort of phishing targets corporate leaders such as CEOs and top-level management employees To infect the target, the fraudster or cyber-criminal employs a URL link.
ADVANTAGES: There is no personalization or customization for the people, this form of attack lacks sophistication. Social information may be included in the email. The recipient's name, company name, designation, friends, co-workers may be missing click the link to accept friend invitations and may even have other people information.

5. Methodology

A phishing website is a social engineering technique that imitates legitimate webpages and uniform resource locators (URLs). The Uniform Resource Locator (URL) is the most common way for phishing assaults to occur. Phisher has complete control over the URL's sub-domains. The phisher can alter the URL because it contains file components and directories. This research used the linear-sequential model, often known as the waterfall model. Although the waterfall approach is considered conventional, it works best in instances where there are few requirements. The application was divided into smaller components that were built using frameworks and hand-written code.
DATASET: We collected the datasets from the open-source platform called Phishing tank. The dataset that was collected was in csv format. There are 18 columns in the dataset, and we transformed the dataset by applying data pre-processing technique. To see the features in the data we used few of the data frame methods for familiarizing. For visualization, and to see how the data is distributed and how features are related to one another, a few plots and graphs are given. The Domain column has no bearing on the training of a machine learning model. We now have 16 features and a target column. The recovered features of the legitimate and phishing URL datasets are simply concatenated in the feature extraction file, with no shuffling. We need to shuffle the data to balance out the distribution while breaking it into training and testing sets. This also eliminates the possibility of over fitting during model training.
Decision Tree Classifier: For classification and regression applications, decision trees are commonly used models. They basically learn a hierarchy of if/else questions that leads to a choice. Learning a decision tree is memorizing the sequence of if/else questions that leads to the correct answer in the shortest amount of time. The method runs through all potential tests to discover the one that is most informative about the target variable to build a tree.
Random Forest Classifier: Random forests are one of the most extensively used machine learning approaches for regression and classification. A random forest is just a collection of decision trees, each somewhat different from the others. The notion behind random forests is that while each tree may do a decent job of predicting, it will almost certainly overfit on some data. They are incredibly powerful, frequently operate effectively without a lot of parameters adjusting, and don't require data scalability.

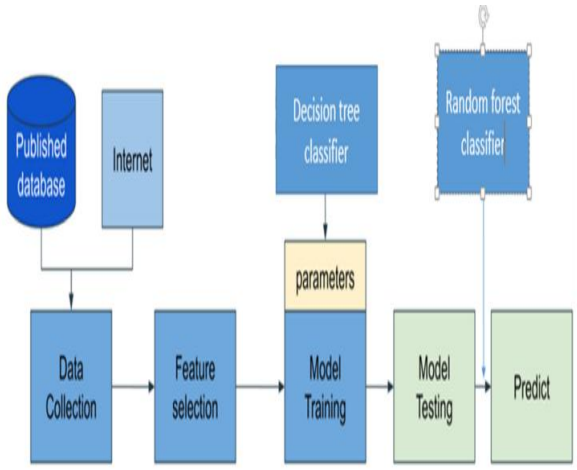

Fig 1: Proposed Architecture Diagram.

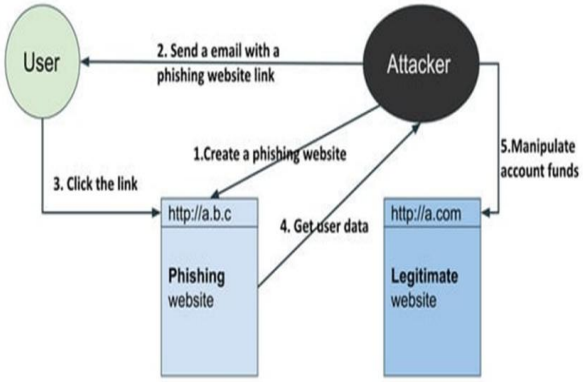

Fig 2: Proposed Flow chart

There's not much mathematics involved here. Since it is very easy to use and interpret it is one of the most widely used and practical methods used in Machine Learning. It is a tool that has applications spanning several different areas. Decision trees can be used for classification as well as regression problems. The name itself suggests that it uses a flowchart like a tree structure to show the predictions that result from a series of feature-based splits. It starts with a root node and ends with a decision made by leaves. Root Nodes – It is the node present at the beginning of a decision tree from this node the population starts dividing according to various features. Decision Nodes – the nodes we get after splitting the root nodes are called Decision Node. Leaf Nodes – the nodes where further splitting is not possible are called leaf nodes or terminal nodes. Sub-tree – just like a small portion of a graph is called sub-graph similarly a subsection of this decision tree is called sub-tree. Pruning – is nothing but cutting down some nodes to stop overfitting.

Important Features of Random Forest Diversity: Not all attributes/variables/features are considered while making an individual tree, each tree is different. Immune to the curse of dimensionality- Since each tree does not consider all the features, the feature space is reduced. Parallelization-Each tree is created independently out of different data and attributes. This means that we can make full use of the CPU to build random forests. Train-Test split- In a random forest we don't have to segregate the data for train and test as there will always be 30% of the data which is not seen by the decision tree. Stability- Stability arises because the result is based on majority voting/ averaging.

PSEUDO CODE FOR PROPOSED METHOD

START

1. System Initialization
   a. Load necessary libraries (Pandas, Scikit-learn, Matplotlib, etc.)
   b. Load the dataset (phishing URLs dataset in CSV format)

2. Data Preprocessing
   a. Read dataset into a data frame
   b. Remove unnecessary columns (e.g., Domain column)
   c. Check for missing values and handle them if necessary
   d. Shuffle the dataset to ensure balanced training
   e. Split dataset into:
      i. Features (X)
      ii. Target (y)

3. Feature Engineering
   a. Visualize feature relationships using graphs (optional)
   b. Normalize or scale features if needed

4. Train-Test Split
   a. Split data into Training set and Testing set (e.g., 80%-20%)

5. Model Training
   a. Initialize machine learning models:
      i. Decision Tree Classifier
      ii. Random Forest Classifier
   b. Train each model on the training data

6. Model Evaluation
   a. Predict on testing data
   b. Calculate metrics:
      i. Accuracy
      ii. Confusion Matrix
      iii. Precision, Recall, F1-score (optional)
   c. Compare models to find the best one (e.g., Random Forest)

7. Save Best Model
   a. Save the best performing model using serialization (e.g., Pickle)

8. Deployment
   a. Create a function that accepts a new URL
   b. Extract features from the URL
   c. Preprocess input features similarly
   d. Use the trained model to predict whether the URL is phishing or legitimate
   e. Display the prediction result

9. End

END

Summary of Main Steps: Input: Dataset of URLs

Processing: Preprocessing + Feature Extraction + Model Training Prediction: Classify URL as "Phishing" or "Legitimate": Output: Model evaluation metrics + Real

6. Results and Discussion

The current system merely detects phishing websites using multiple machine learning techniques and calculates their accuracy. The best model for detecting phishing websites is generated in the suggested system, and the model is saved and deployed, which takes the URL and predicts whether it is a criminal identity theft website or a real website. When compared to the old approach, the aforementioned statements show that this delivers better accuracy in detecting phishing websites. The accuracy of Logistic Regression is 96.63 percent, and the overall comparison is presented.

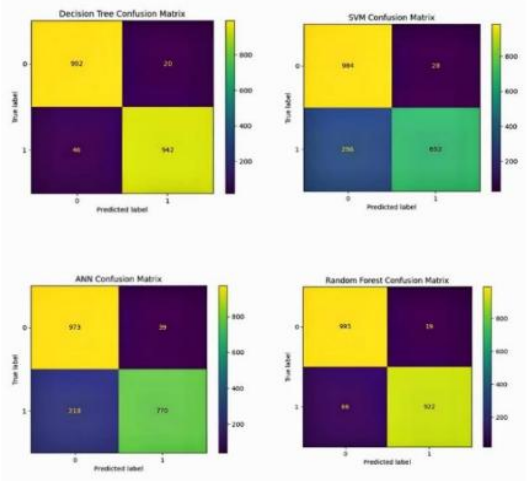

Fig 3. Confusion Matrices for Various Classification Models

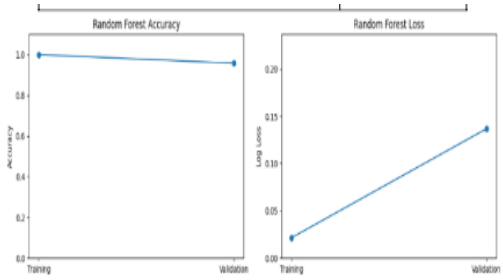

Fig 4. Accuracy and Loss for The Random Forest Model

7. Conclusion

This survey presented various algorithms and approaches to detect phishing websites by several researchers in Machine Learning. On reviewing the papers, we came to a conclusion that most of the work done by using familiar machine learning algorithms like Naïve Bayesian, SVM, Decision Tree and Random Forest. Some authors proposed a new system like Phish Score and Phish Checker for detection. The combinations of features with regards to accuracy, precision, recall etc. were used. Experimentally successful techniques in detecting phishing website URLs were summarized as phishing websites increases day by day, some features may be included or replaced with new ones to detect them.

References

[1] ‘APWG | Unifying The Global Response To Cybercrime’ (n.d.) available: https://apwg.org/

[2] 14 Types of Phishing Attacks That IT Administrators Should Watch For [online] (2021) https://www.blog.syscloud.com,available:https://www.blog.syscloud.comtypes-o f-phishing/

[3] Lakshmanarao, A., Rao, P.S.P., Krishna, M.M.B. (2021) ‘Phishing website detection using novel machine learning fusion approach’, in 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), Presented at the 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), 1164–1169

[4] H. Chapla, R. Kotak and M. Joiser, "A Machine Learning Approach for URL Based Web Phishing Using Fuzzy Logic as Classifier", 2019 International Conference on Communication and Electronics Systems (ICCES), pp. 383-388, 2019, July

[5] Vaishnavi, D., Suwetha, S., Jinila, Y.B., Subhashini, R., Shyry, S.P. (2021) 'A Comparative Analysis of Machine Learning Algorithms on Malicious URL Prediction', in 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), Presented at the 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), 1398–1402

[6] Microsoft, Microsoft Consumer safety report. https://news.microsoft.com/en-sg/2014/02/11/microsoft-consumersafety-index-revealsimpact-of-poor-online-safety-behaviours-in-singapore/sm.001xdu50tlxsej410r11kqvks u4nz.

[7] Internal Revenue Service, IRS E-mail Schemes. Available at https://www.irs.gov/uac/newsroom/consumers-warnedof-new-surge-in-irs-email-schem es-during-2016-tax-season-tax-industry-also-targeted.

[8] Abu-Nimeh, S., Nappa, D., Wang, X., Nair, S. (2007), A comparison of machine learning techniques for phishing detection. Proceedings of the Anti-phishing Working Groups 2nd Annual ECrime Researchers Summit on - ECrime '07. doi:10.1145/1299015.1299021.

[9] E., B., K., T. (2015)., Phishing URL Detection: A Machine Learning and Web Mining-based Approach. International Journal of Computer Applications,123(13), 46-50. doi:10.5120/ijca2015905665.

[10] Wang Wei-Hong, L V Yin-Jun, CHEN Hui-Bing, FANG Zhao-Lin., A Static Malicious Javascript Detection Using SVM, In Proceedings of the 2nd International Conference on Computer Science and Electrical Engineering (ICCSEE 2013).