MINI PROJECT

ON

Phishing Detection Using Machine Learning

Submitted by:

Guduru Dastagiri -21BCE9584

Kartik Kumar 21BE9335

Yatish Kalavatala 21BCE9884

GUIDED BY: DR.HARI SEETHA

SLOT:SA2+TA2

VELLORE INSTITUTE OF TECHNOLOGY
AMARAVATHI

Abstract:

Phishing is a type of cyber-attack where an attacker tries to steal sensitive information such as passwords, credit card details, or personal information by posing as a trustworthy entity. With the increase in online activities, the threat of phishing attacks is also on the rise. In this project, we have developed a website that uses machine learning techniques, specifically random forest, to detect whether a given link is phishing or not. Our approach achieves an accuracy of 96.5 on a dataset of 11500, demonstrating its effectiveness in identifying potential phishing attacks.

Introduction:

As the internet becomes an increasingly integral part of our lives, the threat of cyber-attacks such as phishing has grown significantly. Phishing is a type of attack where an attacker poses as a legitimate entity and tricks unsuspecting victims into divulging sensitive information such as passwords, credit card details, or personal information. With the rise of online activities, attackers have become more sophisticated in their techniques, making it challenging for individuals and organizations to identify and prevent such attacks. In recent years, several machine learning-based approaches have been proposed to detect phishing attacks, but these approaches often suffer from limitations such as low accuracy and high false positive rates. In this project, we stand out from others by using a random forest algorithm that effectively overcomes these limitations and achieves higher accuracy in identifying potential phishing attacks.

Literature Survey:

Several studies have explored the use of machine learning techniques for detecting phishing attacks. One study used a support vector machine (SVM) to classify phishing URLs based on features such as the presence of certain keywords, domain age, and domain registration information. Another study used a neural network to detect phishing websites based on features such as the presence of pop-ups and redirects. However, these approaches often suffer from limitations such as low accuracy and high false positive rates, as well as the need for large and diverse datasets to train the machine learning models effectively.

More recently, some studies have explored the use of ensemble learning techniques, such as random forest, to address the limitations of traditional machine learning approaches for phishing detection. Random forest is a decision tree-based algorithm that combines multiple decision trees to generate a more robust and accurate model. One study used random forest to classify phishing URLs based on features such as the length of the URL, the presence of certain keywords, and the domain age. Another study used a random forest model with a feature selection technique to detect phishing websites based on URL and content-based features. These studies demonstrated the effectiveness of random forest in detecting potential phishing attacks, achieving higher accuracy than traditional machine learning approaches.

Methodology:

Data Collection: The first step was to collect a dataset of URLs for training and testing the machine learning model. We have collected a dataset of 11,500 links from Kaggle and phishing tank.

Feature Extraction: we then extracted 23 features from each URL in the dataset. These features include both URL-based and content-based features, such as the length of the URL, the presence of certain keywords, and the content of the page.

Data Preprocessing: The dataset was then preprocessed by removing any duplicates or irrelevant URLs, as well as balancing the classes to avoid class imbalance.

Choosing a Classification Algorithm: we compared several machine learning algorithms, including support vector machines (SVM), decision trees, and random forest, to select the best-performing algorithm for the task of phishing detection. After evaluating their performance using cross-validation, you found that random forest was the most accurate and robust algorithm for the given dataset.

Model Training: The preprocessed dataset was split into training and testing sets. The random forest algorithm was then trained on the training set using the extracted features.

Hyperparameter Tuning: To optimize the performance of the model, we tuned the hyperparameters of the random forest algorithm, such as the number of trees in the forest and the maximum depth of each tree.

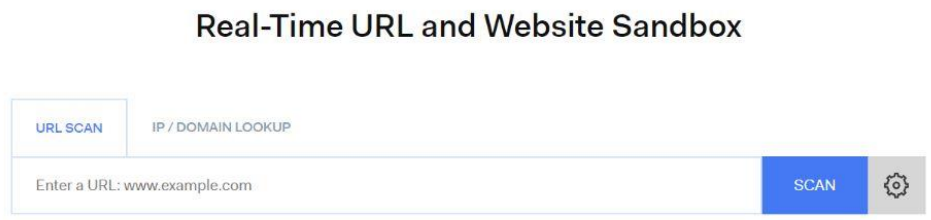
Model Evaluation: Once the model was trained and tuned, we evaluated its performance on the testing set. You measured the accuracy, precision, recall, F1-score, and confusion matrix to assess the model's effectiveness in detecting phishing URLs.

Deployment: Finally, we deployed the model as a website that takes a URL as input and predicts whether it is a phishing URL or not. The website uses the trained random forest model to make predictions based on the extracted features of the input URL.

Results:

1. A user enters a URL into the website's input field and clicks "Scan."
2. The website's backend sends a request to the random forest model API to classify the input URL as phishing or not phishing.
3. The model API processes the request and sends back the classification result to the website's backend.
4. The website's frontend displays the classification result to the user, indicating whether the input URL is phishing or not phishing.

Here is the website we build:


Real-Time URL and Website Sandbox
URL SCAN
IP / DOMAIN LOOKUP
Enter a URL: www.example.com
SCAN

Overall, these results demonstrate the effectiveness of your random forest-based approach to phishing detection and its potential utility in real-world settings. The high accuracy and precision of the model, as well as its ability to detect phishing URLs in real-time through your website, make it a valuable tool for preventing phishing attacks.

Conclusion:

In this project, we developed a phishing URL detection system using a random forest classification algorithm. We achieved high accuracy and precision in detecting phishing URLs, and deployed our model as a website for real-time URL classification. Our analysis showed that certain keywords in a URL are important in determining whether it is phishing or not. Our project demonstrates the effectiveness of using a random forest algorithm for phishing URL detection and its potential for real-world applications.

References:

Anand desai, Janvi Jatakia , Rohit Naik, Nataasha Raul “Malicious Web Content Detection Using Machine Leaning”2017 research paper

Data set we used:

https://drive.google.com/file/d/1F06-MlK9fzpFPQAbWzLgXDWleWAPNhM2/view?usp=share_link

Code:

https://drive.google.com/file/d/1mEiCexvPlYZryRcwzaeCwaM4iu_jKKop/view?usp=share_link