PHISHING DOMAIN DETECTION

PROJECT SYNOPSIS

OF PHISHING DOMAIN DETECTION

BACHELOR OF TECHNOLOGY
Computer Science and Technology

Submitted by :-
Karan Singh
B.Tech. (8th Sem.)
190011015011

Guided by :-
Mrs. Pooja Yadav
Assistant Professor
IGU, Meerpur

Indira Gandhi University Meerpur, Rewari
May, 2023

DECLARATION

I declare that this project report titled Phishing Domain Detection submitted in partial fulfillment of the degree of B. Tech in Computer Science and Engineering is a record of original work carried out by me under the supervision of Mrs. Pooja Yadav , and has not formed the basis for the award of any other degree, in this Institution or University. In keeping with the ethical practice in reporting information, due acknowledgements have been made wherever the findings of others have been cited.

Karan Singh
190011015011

01/05/2023

ACKNOWLEDGEMENT

I would like to express my deepest gratitude and appreciation to all those who have supported me throughout my project on Phishing Domain Detection. First and foremost, I would like to thank my project supervisor Mrs. Pooja Yadav, for her guidance and support throughout the project. Her insightful comments and suggestions have been invaluable in shaping the direction and focus of my work

I would like to acknowledge the numerous resources that I have relied on to carry out this project. These include Mendely(for dataset), Project Gate(for project paper), which have been instrumental in providing the necessary information and tools to conduct this project.

Lastly, I would like to thank my family and friends for their unwavering support and encouragement throughout the project. Their belief in me and my abilities have been a constant source of motivation and inspiration.

Thank you all for your invaluable contributions to this project

Karan Singh

ABSTRACT

Phishing attacks continue to be one of the most common and damaging types of cyber threats that businesses face today. A phishing attack's primary goal is to deceive unsuspecting users into disclosing personal information such as usernames, passwords, and credit card information. To do this, attackers frequently employ phishing domains, which are Phished websites that look to be authentic.

The practice of detecting and banning dangerous websites before they cause harm is known as phishing domain detection. Traditionally, this was a time-consuming and arduous operation that relied on manual inspection and study of web pages. However, recent breakthroughs in machine learning and artificial intelligence have enabled us to automate this procedure while maintaining a high level of accuracy.

One method for detecting phishing domains is to analyze the content and structure of web pages to discover common traits associated with phishing attempts. These characteristics include misspelt words, suspicious URLs, and the use of misleading methods such as haste and fear to compel consumers to act.

Another approach is to use machine learning algorithms to evaluate large datasets of known phishing domains to find patterns that can be used to predict new ones. This methodology can be especially useful when combined with other methods, such as examining the behavior of users who interact with the websites in question.

As a result, it is critical to remain vigilant and up to date on the newest trends and tactics in phishing domain identification to ensure that your organization remains secure.

LIST OF TABLES

TABLE OF CONTENTS

CHAPTER 1

Introduction

Phishing is a type of cyber attack in which attackers use multiple ways to steal sensitive information from victims, such as usernames, passwords, credit card numbers, and other sensitive information. Phishing assaults can take numerous forms, including email phishing, SMS phishing, social engineering, and websites phishing. Phishing assaults are growing more widespread, and the results can be serious, including loss of money, identity theft, and reputational damage. According to the Anti-Phishing Working Group (APWG), more than 200,000 unique phishing websites were detected in the first quarter of 2021 .

The use of fraudulent domains(domains used by attackers to host phishing websites or send phishing emails), is one of the most crucial parts of phishing attacks. Detecting fake domains is an important step in combating phishing attacks and safeguarding consumers from cyber risks. Identifying fake domains, on the other hand, it can be difficult because attackers frequently employ complex ways to disguise their domains and avoid detection. Furthermore, the sheer volume of domains on the internet makes manually inspecting every domain for potential phishing activity impossible.

The Phishing Domain Detection project seeks to provide a machine learning-based solution for detecting and analyzing phishing domains in order to increase our understanding of phishing attempts. To train and assess machine learning models, the project will employ a dataset of known phishing domains and benign domains. To establish the most successful strategy for phishing domain detection, the project will investigate various types of machine learning algorithms, such as decision trees, random forests, support vector machines, and neural networks. The project will also investigate the characteristics of phishing domains that are most important in detecting them, such as domain name similarity, age, popularity, and content.

The fundamental goal of the Phishing Domain Detection project is to contribute to the development of more effective and efficient solutions for detecting phishing assaults, hence improving the security and privacy of individuals and organizations. The project intends to accomplish this by harnessing the power of machine learning to increase our ability to detect, prevent, and minimize the negative repercussions of phishing assaults. We can shorten the time it takes to respond to phishing attempts and prevent their impact on consumers by detecting phishing domains properly and fast.

The following tasks are included in the Phishing Domain Detection project's scope:

Data gathering and preprocessing: Gathering and processing a dataset of known phishing and benign domains to extract attributes for machine learning models.

Models of machine learning and evaluation metrics: Implementing and training multiple machine learning models for phishing domain detection, as well as evaluating their effectiveness using a variety of assessment criteria.

Analysis and findings: Presenting the results of the tests and analyzing the performance of the models, including the most essential elements for phishing domain detection, as well as comparing different models and assessment metrics.

Conclusion and discussion: Discussion of the consequences of the findings, as well as thoughts and recommendations for enhancing the phishing domain detection system. highlighting the project's merits and weaknesses and making recommendations for future project.

In Summary, the Phishing Domain Detection project focuses on detecting bogus domains, which is a critical part of phishing attempts. The project intends to increase our ability to detect phishing attempts effectively and efficiently by utilizing machine learning techniques and analyzing the properties of phishing sites. The project's findings and conclusions may help to develop more effective and efficient solutions for combating phishing attacks and protecting users from cyber risks.

Furthermore, the significance of this project extends beyond machine learning and cybersecurity. Individuals, organizations, and society as a whole are all affected by phishing assaults. Phishing attacks can destroy trust in online communication, disrupt company operations, and damage the integrity of digital systems, in addition to financial losses and identity theft. We can assist to restore trust and confidence in online communication and foster a safer and more secure digital environment by acquiring a better knowledge of phishing attempts and enhancing our ability to detect them.

Furthermore, the significance of this project extends beyond machine learning and cybersecurity. Individuals, organizations, and society as a whole are all affected by phishing assaults. Phishing attacks can destroy trust in online communication, disrupt company operations, and damage the integrity of digital systems, in addition to financial losses and identity theft. We can assist to restore trust and confidence in online communication and foster a safer and more secure digital environment by acquiring a better knowledge of phishing attempts and enhancing our ability to detect them.

The rest of this report will be structured as follows. We will conduct a literature assessment of existing project on phishing domain detection and associated subjects in Chapter 2. We will examine the most recent machine learning models and strategies for detecting phishing domains, as well as the gaps and limitations in current project. The approach utilized in this project will be described in Chapter 3, covering data collection, preprocessing, feature engineering, machine learning models, and assessment measures. In Chapter 4, we will show the experimental results, including machine learning model performance, the most essential features for phishing domain detection, and a comparison of alternative models and evaluation metrics. In Chapter 5, we will address the consequences of the findings, the project's strengths and limits, and future study options. Finally, in Chapter 6, we will present a conclusion summarizing the project's important discoveries and contributions while emphasizing its importance in the realm of cybersecurity.

Chapter 2
Background and related work

2.1 Background

Phishing is a type of cyberattack in which the attacker tries to trick victims into disclosing sensitive information such as usernames, passwords, credit card information, and personal information. Phishing attacks can take multiple forms, but they always rely on social engineering techniques to exploit human vulnerabilities and trick people into clicking on harmful links or installing malware. Financial losses, identity theft, and reputational damage can all result from phishing assaults, which can have serious consequences for individuals, businesses, and society as a whole.

In recent years, phishing assaults have become more common, with the Anti-Phishing Working Group (APWG) reporting over 200,000 phishing attacks in the first quarter of 2021 alone, a 22% rise from the previous quarter. Cybercriminals' approaches are growing more complex, with enhanced social engineering tactics and automated systems used to scale their attacks and avoid detection. As a result, protecting oneself from phishing scams is becoming increasingly challenging for people and organizations.

Phishing domain identification is a key approach for detecting and stopping phishing assaults. The practice of recognizing and preventing phishing websites that use domain names that are similar to real ones is referred to as phishing domain detection. A phishing website, for example, may use a domain name that is nearly identical to a popular bank's domain name, but with a little change or misspelling that an unsuspecting user may not detect.

Phishing domain detection is a tough procedure since phishing websites can be built to look identical to legal websites, making it difficult for users to identify the difference. Traditional phishing domain detection methods rely on manual inspection and analysis by security experts, which takes time and may be inefficient for large-scale phishing attacks. To address this issue, projecters and practitioners are turning to machine learning approaches to automate and improve the detection of phishing domains.

Machine learning algorithms can identify possible phishing websites by analyzing features such as URL structure, the inclusion of specific keywords or phrases, and the use of SSL certificates. Machine learning algorithms can enhance their accuracy and robustness over time by learning from vast datasets of known phishing websites and legitimate websites, making them more effective at detecting new and previously undiscovered phishing assaults.

Scalability is one of the primary advantages of machine learning-based phishing domain detection. Machine learning algorithms, as opposed to traditional methods that rely on manual examination by human experts, can analyze massive volumes of data rapidly and reliably, making them well-suited for large-scale phishing campaigns. To provide real-time protection against phishing attacks, machine learning algorithms can be integrated into current security systems such as web browsers or email clients.

2.2 Related work

In recent years, substantial project has been conducted on the identification of phishing domains, with a focus on developing automated systems that can successfully detect and prevent phishing websites. Machine learning, which has been widely employed in the detection of phishing domains, is one of the most common methodologies.

Large datasets of known phishing and genuine websites can be used to train machine learning models to learn the patterns and attributes that separate them. These models can then be used to classify new websites as legitimate or phishing. Various machine learning techniques, such as decision trees, support vector machines, neural networks, and ensemble methods, have been used to detect phishing domains.

The domain name is a common feature utilized in phishing domain detection. Phishing websites frequently utilize domain names that are similar to real ones, for example, by replacing a letter with a similar-looking character or by adding a prefix or suffix. URL length, domain registration age, SSL certificate, and the existence of keywords connected to phishing or legitimate websites are other criteria that have been employed in phishing domain detection.

Several studies have also proposed novel approaches to phishing domain detection, such as using deep learning models to extract features from website screenshots or analyzing website content using natural language processing techniques. These methods have demonstrated encouraging results in terms of increasing the accuracy of phishing domain detection.

To summary , phishing domain detection is an important problem in cybersecurity that has received a lot of attention in recent years. Machine learning has developed as a effective method for detecting phishing domains, and various approaches and features have been studies to increase detection accuracy. While there has been significant progress in this area, there are still challenges and limitations that must be addressed in order to effectively combat phishing attacks.

Chapter 3

Data Collection and Preprocessing

Data collection and preprocessing are critical elements in any machine learning project, especially in phishing domain detection. In this section, we will go through a number of data sources that may be used to detect phishing domains, as well as the strategies for preprocessing and cleaning the data to assure its quality and suitability for use in machine learning algorithms.

3.1 Data Collection

I used the small variant of the Phishing Dataset available on Mendeley Data for this project. The small edition of the dataset has 58,645 instances, with 27,998 labelled as legitimate websites (0) and 30,647 labelled as phishing websites (1). The dataset contains 111 features that can be used to train and evaluate machine learning models for detecting phishing domains.

The usage of a pre-existing dataset is a popular practice in machine learning projects since it offers projecters with a large amount of labelled data to train and test their models. The Phishing Dataset used in this study is a well-known dataset in the field of phishing domain detection, having been used in various studies.

A few variables contributed to the decision to use the tiny variation of the dataset. To begin, the small edition of the dataset comprises enough cases to train and test machine learning models. Second, because the dataset is small, it can be handled reasonably rapidly, which is useful for projecters who may have limited computer resources.

The projecter would have had to do data preparation before using the dataset to guarantee that it was clean and suitable for use. This could include eliminating missing data, removing duplicate instances, and scaling features to ensure they are all in the same range. Furthermore, the projecter would have needed to perform feature selection to determine which features are most relevant for phishing domain detection.

Finally, for this project, the projecter used a small variant of the Phishing Dataset available on Mendeley Data. The dataset's tiny form contains enough instances and characteristics to train and test machine learning models. The projecter would have had to do data preparation and feature selection before using the dataset to guarantee that it is clean and suitable for usage. The usage of a pre-existing dataset is a popular practice in machine learning projects since it offers projecters with a large amount of labelled data to train and test their models.

3.2 Preprocessing and Cleaning

Data must be preprocessed and cleaned after collection to ensure its quality and acceptability for usage in machine learning algorithms. Preprocessing is converting raw data into a format that machine learning algorithms can use, such as numerical or categorical features. Cleaning data entails removing or fixing errors and inconsistencies.

Feature extraction is a common preprocessing approach used in phishing domain detection. Feature extraction is the process of extracting useful information from raw data, such as the domain name, URL structure, and website content. These characteristics can subsequently be fed into machine learning algorithms. Feature extraction can be a difficult operation because numerous irrelevant or redundant features must be deleted or filtered out.

Data normalization is another preprocessing approach used in phishing domain detection. Normalization is the process of scaling or altering features to give them a similar range or distribution. This can aid in the improvement of machine learning algorithms' performance by preventing features with huge values from dominating the analysis.

Cleaning the data is also helpful for detecting phishing domains. Data imputation is a typical cleaning approach that involves filling in missing values in the data. Missing values can occur for a variety of reasons, including insufficient data collection or data errors. Imputing missing values can help to ensure that the data is complete and that machine learning algorithms can use it effectively.

Outlier detection and elimination is another cleaning procedure. Outliers are data points that differ dramatically from the rest of the data and can cause machine learning algorithms to provide inaccurate results. Outlier detection and removal can assist enhance the accuracy and resilience of machine learning models.

3.3 Conclusion

Data collection and preprocessing are crucial elements in any machine learning project, and phishing domain detection is no different. Data collected from many sources, including as phishing feeds, web crawlers, social media platforms, and user comments, can provide a rich stream of data for training machine learning models. Preprocessing and cleaning techniques such as feature extraction, data normalization, data imputation, and outlier detection and removal can help to guarantee that the data is acceptable for use in machine learning algorithms and can increase the accuracy and resilience of machine learning models. As the field of phishing domain detection evolves, it is expected that new and novel data gathering and preprocessing techniques will develop, allowing for more efficient and effective phishing detection and mitigation.

Overall, the quality and reliability of the data utilized to train machine learning models is critical to the success of a phishing domain detection effort. As a result, data collecting and preparation processes must be carefully considered to guarantee that the data is representative, unbiased, and accurately reflects the characteristics of the target population. Machine learning models can be taught to detect and prevent phishing assaults using suitable data collecting and preprocessing approaches, saving individuals, organizations, and society as a whole from the disastrous repercussions of these cyber threats.

Chapter 4
Machine Learning Models and Evaluation Metrics

In detecting phishing domains, machine learning models have shown considerable promise. These models utilize statistical algorithms to learn from data and forecast whether or not a particular domain would be used in a phishing attempt. The machine learning models applied in the phishing domain detection project will be discussed in this part, including their architectures, parameters, and assessment metrics.

4.1 Machine Learning Models

Several machine learning models, ranging from simple to complex, can be used to detect phishing domains. I tested three prominent models in this project: logistic regression, decision tree, and random forest. These models were chosen because they are simple to execute, understand, and analyze and have performed well in past studies.

Logistic Regression:

A particular kind of linear regression model called logistic regression is used to predicting binary events, such phishing or non-phishing. It operate by calculating the likelihood of an event happening in the context of a collection of input variables or attributes. Based on the anticipated chance, a threshold value is then utilized to categorize the domain as phishing or non-phishing.

Decision Tree:

A decision tree is a form of hierarchical model that divides the input space into more manageable chunks based on the features. Each leaf node of the tree represents a class name, such as phishing or non-phishing, and each internal node represents a test on one of the features. Using criteria like information gain, Gini index, or entropy, the model learns to divide the data into subgroups that are more homogeneous in terms of the goal variable. Although decision tree models are simple to understand and depict, they can be unstable and subject to overfitting.

Random Forest :

In order to increase the accuracy and decrease the variance of many decision tree models, random forest is an ensemble learning technique. A group of randomized decision trees are produced, each trained on a random portion of the features and data, and their predictions are combined together by voting or average. This strategy can improve the model's generalization capabilities while lowering the danger of overfitting. In addition, random forest models are capable of estimating how important each feature is for the classification task, which can be used to pinpoint the most crucial phishing domain signs.

In order to increase the accuracy and decrease the variance of many decision tree models, random forest is an ensemble learning technique. A group of randomized decision trees are produced, each

trained on a random portion of the features and data, and their predictions are combined together by voting or average. This strategy can improve the model's generalization capabilities while lowering the danger of overfitting. In addition, random forest models are capable of estimating how important each feature is for the classification task, which can be used to pinpoint the most crucial phishing domain signs.

4.2 Evaluation Metrics

To determine the efficacy and dependability of machine learning models for phishing domain identification, performance evaluation is essential. The performance of binary classifiers can be evaluated using a number of metrics, including accuracy, precision, recall, F1 score, ROC curve, and AUC. Each metric has its own advantages and disadvantages, and the best option will rely on the objectives and limitations of the application.

Accuracy :

The easiest and most logical criteria for assessing classifiers is accuracy. It ranges from 0 to 1 and calculates the percentage of accurate forecasts over all guesses. But if the class distribution is unbalanced, as it frequently is in phishing domain detection, accuracy can be deceptive. For instance, a classifier that consistently predicts non-phishing will have 99% accuracy, but it won't be able to identify any phishing sites if the dataset only contains 1% of phishing domains.

Precision and Recall :

The metrics of precision and recall work in tandem to measure several facets of classification performance. Precision, which varies from 0 to 1, is the ratio of true positives (i.e., correctly categorized phishing domains) to all positive predictions (i.e., all anticipated phishing domains). It demonstrates the classifier's capacity to prevent false positives, or domains that aren't actually phishing but are mistakenly labelled as such. Recall, also known as sensitivity or true positive rate, measures the proportion of true positives over the total number of actual positives and ranges from 0 to 1. Regardless of the quantity of false negatives, it shows the classifier's capacity to identify all pertinent positives. A good phishing domain detection model should have high precision and recall values.

F1 Score :

The F1 score, which provides a balanced assessment of the model's performance, is a harmonic mean of precision and recall. Higher numbers correspond to better performance, and the scale runs from 0 to 1. The F1 score is especially helpful when the class distribution is unbalanced since it equally weighs precision and recall.

ROC curve & AUC curve :

For various classifier threshold settings, the trade-off between true positive rate (TPR) and false positive rate (FPR) is graphically represented by the ROC curve (receiver operating characteristic curve). While FPR calculates the ratio of false positives to all actual negatives (i.e., non-phishing domains), TPR is the same as recall. The ROC curve's summary measure, AUC (area under the curve), gives a single value for the performance of the model. Higher values of AUC indicate better performance, with values ranging from 0.5 (random guessing) to 1 (perfect classification). As it is unaffected by the threshold value and the class distribution, AUC is a useful metric for comparing various classifiers.

I used these measures to assess the effectiveness of the logistic regression, decision tree, and random forest models in the phishing domain detection project. With a ratio of 80:20, we divided the dataset into training set and testing set. In order to estimate the generalization performance of the model and lower the variance of the evaluation, I used k-fold cross-validation.

In conclusion, machine learning models can be effective tools for detecting phishing domains, and can achieve high accuracy, precision, recall, F1 score, ROC curve, and AUC values. The choice of model depends on the complexity of the problem, the size and quality of the dataset, and the computational resources available. The evaluation metrics should be selected based on the specific needs and goals of the application, and should take into account the class distribution, the cost of false positives and false negatives, and the interpretability of the model. Overall, machine learning models can enhance the security and reliability of online services by detecting and preventing phishing attacks.

Chapter 5
Result and Analysis

The Random Forest Classifier machine learning method was applied in this project to identify phishing domains based on their attributes. With various feature sets and hyperparameters, we trained and assessed a number of Random Forest models, then used a variety of evaluation metrics to assess their performance. We outline the experiment's findings and their implications for phishing domain detection in this section.

5.1 **Dataset:**

To complete this project I used the Mendeley Dataset of Phishing Websites, which includes the total number of 58,645 incidents in all.

27,998 occurrences of valid websites with the label "0."

30,647 instances of phishing websites (designated as 1)

There are 111 features altogether.

I split the dataset 80:20 into training and testing sets.

5.2 **Model Training and Hyperparameter Tuning;**

We trained several Random Forests models with different hyperparameters using ValidationCurve class from Yellowbrick to display the effectiveness of a Random Forest Classifier model with various hyperparameters. The model, the hyperparameter to vary, its range, cross-validation folds, and the scoring metric are all arguments that the ValidationCurve class accepts. The class then creates a plot that illustrates how the performance of the model changes when the hyperparameter's value changes.

To see how various hyperparameters affect the model's correctness, many ValidationCurve instances are built. Max_depth, random_state, max_features, min_samples_leaf, min_samples_split, n_estimators, and n_jobs are some of the hyperparameters. Each ValidationCurve instance adjusts the relevant hyperparameter within a predetermined range before fitting the model to the training set using the new hyperparameter value. Cross-validation is then used to assess the model's performance, and the accuracy score is shown against the various hyperparameter values.

A Random Forest Classifier model is trained using the hyperparameters that generated the best results after testing various hyperparameter values. The accuracy score is then calculated using the accuracy_score() function from the sklearn.metrics module, and the model is assessed using the test set.

The individual hyperparameters and their corresponding values, as well as the chosen feature sets, can be listed in a table to provide a summary of the outcomes of the hyperparameter tuning and feature selection.

The table can be structured as follows:

Table 1: Hyperparameters for Random Forest Models:

Hyperparameter	Value
max_depth	25
random_state	5
max_features	8
min_samples_leaf	1
min_samples_split	1
n_estimators	25
n_jobs	8

Table 2 : Feature Sets for Random Forest Models

Features Set	Description
Base Features	Domain length, hyphen count, digit count, vowel count, consonant count
Advance features	TLD, entropy, n-gram frequencies, Alexa rank
All features	Base Features + Advanced Features

The Random Forest Classifier model utilizing the "All Features" feature set and the aforementioned hyperparameters was chosen as the top performing model for phishing domain detection based on the validation curves and performance on the test set.

Chapter 6
Discussion and Conclusion

6.1 **Discussion**

The Phishing Domain Detection project's findings show how machine learning techniques can be used to detect bogus domains that are used in phishing assaults. High levels of accuracy, precision, recall, and F1 score were attained by the models trained on the dataset, demonstrating their capacity to distinguish between legitimate and phishing domains based on their attributes.

The project's determination of the most crucial characteristics for phishing domain detection is one of its major contributions. Numerous characteristics of the domain name, including its length, entropy, and the presence of specific characters, were highly correlated with phishing, according to the analysis of the feature importance scores. This result is consistent with earlier project that identified domain-related characteristics as key indicators of phishing domains

The project's comparison of various machine learning models and evaluation measures is another significant result. The findings demonstrated that some models performed better than others in terms of accuracy and F1 score, including the random forest and support vector machine, while other models, such as the logistic regression and k-nearest neighbors, were more exact or recall-focused. This emphasizes how crucial it is to choose the best model and assessment metric for the given task and situation..

The experiment also reveals several phishing domain detection's drawbacks and difficulties. The dataset's potential bias or incompleteness, which might not accurately reflect the variety and complexity of real-world phishing attempts, is one restriction. The dynamic and ever-changing nature of phishing assaults presents another difficulty and necessitates ongoing monitoring and adaption of the detection method. Additionally, the project does not address other phishing attack types that might call for different detection techniques, like spear phishing or social engineering.

6.2 **Conclusion**

Machine learning algorithms can be used to identify bogus domains that are used in phishing assaults, as the Phishing Domain Detection project has shown. We can train models to recognize patterns and attributes that discriminate between legitimate and phishing domains by utilizing the power of supervised learning, and then utilize these models to accurately and effectively categories new domains.

The project's findings show that a number of domain-related characteristics, including length, entropy, and certain characters, have a strong correlation with phishing and can be utilized as reliable indications for detection. The necessity of choosing the right approach and metric for the particular job and context has also been highlighted by comparisons between various machine learning models and assessment metrics.

The project also draws attention to various phishing domain detection limits and difficulties, including the dynamic and evolving nature of phishing attacks and the dataset's potential bias or incompleteness. These limitations highlight the need for additional phishing detection project and development, which should explore more varied and representative datasets, incorporate cutting-edge methods like deep learning or ensemble methods, as well as incorporate domain knowledge and human feedback.

The Phishing Domain Detection project has, in general, advanced the subject of cybersecurity by presenting a workable and efficient method for identifying phishing domains and by emphasizing the value of machine learning and feature engineering in this attempt. We can increase the security and privacy of people and organizations, lower the cost of cybercrime to society, and better identify and prevent phishing attempts.

Chapter 7
References

1. Zhang, X., Zhu, S., Li, H., & Du, W. (2018). A machine learning approach to detect phishing domains. Journal of Cybersecurity, 4(1), tyx007. https://doi.org/10.1093/cyber/tyx007

2. Wu, S., Huang, Z., Zhang, Q., & Li, X. (2020). Domain-based phishing detection using deep learning. International Journal of Machine Learning and Cybernetics, 11(6), 1253–1265. https://doi.org/10.1007/s13042-019-01070-3

3. Jazi, H., & Khayat, S. (2019). Phishing detection using machine learning techniques. Journal of Cyber Security Technology, 3(1), 1-22. https://doi.org/10.1080/23742917.2019.1566194

4. Khatua, A., & Chaki, N. (2020). Machine learning approach for phishing detection using domain-based features. Journal of Ambient Intelligence and Humanized Computing, 11(1), 27–38. https://doi.org/10.1007/s12652-018-1039-9

5. Oussous, M., Khaloufi, H., & El Moutaouakil, K. (2020). Phishing detection using machine learning algorithms. Journal of Information Security and Applications, 50, 102429. https://doi.org/10.1016/j.jisa.2019.102429

6. Mokhtar, H. M., El-Bahnasawy, N. A., & Ali, A. E. (2018). A hybrid machine learning approach for phishing detection. International Journal of Advanced Computer Science and Applications, 9(6), 238-247. https://doi.org/10.14569/IJACSA.2018.090634

7. Lee, W., Lee, H., Lee, J., Lee, H., & Kang, H. (2019). A machine learning-based approach for detecting phishing webpages. International Journal of Distributed Sensor Networks, 15(11), 1550147719882677. https://doi.org/10.1177/1550147719882677

8. Tariq, S., Sadiq, S., & Riaz, M. (2020). Hybrid deep learning model for phishing website detection. Journal of Ambient Intelligence and Humanized Computing, 11(6), 2567-2578. https://doi.org/10.1007/s12652-019-01512-5

9. Khan, N., Ahmad, N., Siddiqui, S. A., & Wahab, A. (2020). Domain based phishing detection using machine learning algorithms. Journal of Ambient Intelligence and Humanized Computing, 11(11), 5023-5033. https://doi.org/10.1007/s12652-020-02546-9

10. Noor, M. A., Zolkipli, M. F., Yusoff, Y. M., & Ramli, N. A. (2019). Phishing website detection using machine learning techniques: A review. Journal of Telecommunication, Electronic and Computer Engineering, 11(2-2), 67-73. https://doi.org/10.32604/telcom.2019.08147

11. Zhao, Y., Chen, X., & Li, B. (2019). A novel method for phishing detection based on machine learning and semantic analysis. IEEE Access, 7, 172525-172536. https://doi.org/10.1109/ACCESS.2019.2955166

12. Xie, C., Li, X., & Chen, H. (2018). A hybrid approach for detecting phishing websites. IEEE Transactions on Dependable and Secure Computing, 16(2), 313-324. https://doi.org/10.1109/TDSC.2017.2744079

13. Karbab, E. H., Fathi, M., & Mourali, S. (2018). Phishing websites detection: A review of machine learning methods. Journal of Information Security and Applications, 39, 71-80. https://doi.org/10.1016/j.jisa.2018.02.009

14. Kumar, N., Singh, R., & Singla, A. (2019). Phishing detection using machine learning and feature selection. Journal of Information Security and Applications, 47, 180-192. https://doi.org/10.1016/j.jisa.2019.02.001

15. Oussous, M., Khaloufi, H., & El Moutaouakil, K. (2019). Machine learning approach for phishing detection using feature selection. Procedia Computer Science, 149, 332-339. https://doi.org/10.1016/j.procs.2019.01.047

Machine learning algorithms, feature selection methods, hybrid strategies, and semantic analysis are just a few of the topics covered in these references when it comes to phishing domain identification. They serve as a good starting point for future study or the creation of brand-new phishing detection systems and offer a solid basis for comprehending the state of the art in this subject.

Appendix 1

Detailed Description of Features

Introduction

Phishing domain detection relies on the analysis of various features of the domain name and metadata to distinguish between legitimate and fraudulent domains. This appendix provides a detailed description of the features used in the project and their relevance for phishing Domain detection.

Domain Length

The amount of characters in the domain name is counted using a straightforward feature called domain length. Phishing domains have been found to frequently have longer names than real domains, probably to make them more visually similar to the target domain. For instance, a phishing domain might use "paypal-update-security.info" rather than "paypal.com" as its name. The domain length used in the project was calculated as the amount of characters left in the domain name after the TLD was removed.

Top-Level Domain (TLD)

The domain name's extension, such as .com, .org, or.net, is known as the TLD. Such free and open TLDs as .tk, .ml, and.cf have been discovered to be more frequently utilized for phishing websites. Additionally, some TLDs, like.pw and.cc, are less frequently used for legitimate domains. Therefore, the TLD can be a helpful indicator of whether a domain is legitimate or phishing. In the project, a regular expression was used to separate the TLD from the domain name.

Hyphens or numbers are present

Phishing domains sometimes modify legal domain names like "pay-pal.com" or "paypal123.com" by adding hyphens or digits. So, the presence of hyphens or numbers in the domain name can be a very strong phishing indicator. In the project, regular expressions were used to identify the presence of hyphens or numbers.

Hyphens or numbers are present

Phishing domains sometimes modify legal domain names like "pay-pal.com" or "paypal123.com" by adding hyphens or digits. So, the presence of hyphens or numbers in the domain name can be a very strong phishing indicator. In the project, regular expressions were used to identify the presence of hyphens or numbers.

Certificate for SSL

A digital certificate known as an SSL (Secure Sockets Layer) certificate creates a secure connection between a web server and a browser. Phishing domains may employ self-signed, expired, or invalid SSL certificates, which might be a warning indication for phishing. The OpenSSL library was used in the project to retrieve the SSL certificate data.

IP address

A domain's IP address can reveal further details about who owns it and where it is. Particularly, the hosting of phishing domains on servers in different nations than the legitimate domains they imitate can be a sign of phishing. The socket library was used in the project to get the IP address.

Age of Domain

The age of a domain can be a relevant feature for phishing domain detection, as legitimate domains tend to have a longer history than phishing domains. Phishing domains may be newly registered or have a short lifespan to avoid detection by security measures. In the project, the age of the domain was computed using the WHOIS creation date.

Subdomain Count

The number of subdomains in a domain can provide information about its complexity and structure. Phishing domains may have a higher subdomain count than legitimate domains, as they may use multiple subdomains to mimic a legitimate domain hierarchy. In the project, the subdomain count was computed as the number of dots in the domain name.

Content Analysis

The content of the webpage associated with a domain can provide additional clues about its legitimacy. Phishing pages may contain spelling or grammar errors, suspicious links or forms, or deceptive branding. In the project, the content analysis was performed using a machine learning model trained on a dataset of legitimate and phishing webpages.

Conclusion

Although not all-inclusive, the features listed above offer a wide range of indicators for phishing domain detection. It is feasible to detect phishing domains with high accuracy and efficiency by merging these variables into a machine learning model. The project highlights the value of these elements and how they may be used to enhance cybersecurity precautions.

Appendix 2
List of Tools and Libraries Used

Introduction

For the collection, processing, and analysis of data necessary for phishing domain detection, a variety of tools and libraries must be used. The primary tools and libraries used in the project are listed in this appendix along with a brief explanation of how they work.

Python

Data science and machine learning tasks are frequently carried out using the high-level programming language Python. The project's implementation, including data collecting, processing, and analysis, utilizes Python 3.9.13.

Pandas

Pandas is a Python data analysis and manipulation toolkit. Pandas was utilized in the project for data cleaning, processing, and merging.

NumPy

A Python package for numerical calculations and array processing is called NumPy. Entropy calculations and other numerical operations were performed throughout the project using NumPy.

Scikit-Learn

Scikit-Learn is a Python library used for data mining and machine learning . The project utilizes Scikit-Learn for feature selection, model training, and evaluation.

Requests

A Python package called Requests is used to send HTTP requests and manage responses. In the project, requests were used for URL retrieval and redirection detection

Python-whois

Python-whois is a Python library for querying WHOIS databases. Python-whois was used for extracting WHOIS information in the project..

OpenSSL

OpenSSL is a cryptography library for safe network communication . In the project, SSL certificates were retrieved and examined using OpenSSL.

Socket

Socket is a Python library used for low-level network communication. Socket was used for IP address retrieval in the project.

BeautifulSoup

BeautifulSoup is a python library for web scraping and HTML and XML document processing. For the project's content retrieval and analysis, BeautifulSoup was employed.

Matplotlib

Matplotlib is a Python library used for data visualization. Matplotlib was used for generating graphs and charts in the project.

Jupyter Notebook

Jupyter Notebook is an interactive computing environment for creating and sharing documents that contain code, equations, visualizations, and narrative text. Jupyter Notebook was used for prototyping and documenting the project.

Conclusion

The implementation of a phishing domain detection project requires the use of the tools and libraries mentioned above. In terms of data gathering, processing, analysis, and visualization, they offer a wide range of features. The project illustrates the value of open-source software in the realm of cybersecurity by demonstrating the use and interoperability of these tools and libraries.