International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 11 Issue: 11 | Nov 2024 www.irjet.net p-ISSN: 2395-0072

Proactive Phishing Page Detection With SafeNet: Enhancing Cybersecurity Measures

[1]Vaishnavi Tate , [2]Shrutika Chavan , [3]Rohini Bagal , [4]Prachi Kolhal , [5]Asmita Vasekar
[6]Dr. Mrs. S. P. Pawar

[1,2,3,4,5] UG Students, Department of Computer Science And Engineering, SVERI's College of Engineering Pandharpur, Maharashtra, India
[6]Assistant Professor, Department of Computer Science And Engineering , SVERI's College of Engineering Pandharpur, Maharashtra, India

---

ABSTRACT

Phishing is one of the most prevalent online threats in today's digital landscape. These attacks typically use fraudulent website URLs to steal sensitive personal information, such as login credentials and credit card details. As technology advances, phishing techniques continue to evolve, becoming more sophisticated and harder to detect.

In this project, we developed a phishing detection system using FastAPI, leveraging machine learning to distinguish between legitimate and malicious URLs. By employing logistic regression and a multimodal neural processing algorithm, we analyzed a curated dataset of malicious links to train and evaluate our model. The primary goal was to determine whether a given URL is safe or a potential phishing threat. This work highlights the potential of machine learning in combating phishing attacks and improving online security.

Keywords: Phishing, Logistic Regression, Machine Learning, Proactive, SafeNet

1. INTRODUCTION

Phishing has become a significant concern for security researchers due to the ease of creating fake websites that closely resemble legitimate ones. While experts can often identify such fraudulent sites, average users may fall victim to phishing attacks. The primary objective of these attacks is to steal sensitive information, such as bank account credentials. For instance, attackers may send deceptive emails claiming that a user's account password is about to expire, prompting them to click a link. This link redirects to a fake page hosted on a hacker's server, where personal data is stolen.

In our project, we aim to predict whether URLs are legitimate or malicious. The dataset includes phishing URLs from the open-source service PhishTank. URLs with no malicious detection were labeled as benign ('0'), while those flagged by at least eight detections were labeled as phishing ('1'). Using machine learning algorithms, we analyze URL characteristics to understand how phishing sites are constructed and identify potential threats.

Phishing attacks thrive due to a lack of user awareness and exploit human vulnerabilities. Attackers employ innovative tactics like obfuscation, fast- flux (dynamic proxy generation), and algorithmic URL generation to evade blacklist defenses.

2. LITERATURE SURVEY

The home service**[3]** industry has experienced rapid growth in recent years, largely driven by the shift from traditional methods to digital platforms that prioritize convenience and ease of access. As the global home services market has expanded significantly, fueled by factors such as urbanization, increased disposable income, and a growing demand for on-demand services. In today's digital era, platforms are becoming the go-to solution for connecting customers with service providers. A key driver behind this growth is the mobile-first approach, particularly through Android apps, which offer users a convenient and accessible way to access services directly from their mobile devices.

1. **Akriti Soni, Pranchal Abrol: Phishing Website Detection, 2022, [15]:** Akriti Soni and Pranchal Abrol proposed that online phishing is one of the most common attacks on the modern internet, aiming to steal personal data like login credentials and credit card numbers. As technology evolves, so do phishing strategies. In their project, they built a phishing detection system using FastAPI, employing logistic regression and multimodal NP algorithms. The goal is to determine if a website URL is good or bad by creating and curating a dataset of malicious links for the machine learning model.

2. **Chunlin Liu, Bo Lang: Finding effective type for malicious URL detection: In ACM, 2018,[3]:** Chunlin et al. focused on individual frequency features and combined statistical analysis of URLs with machine learning, achieving 99.7% precision and a false positive rate below 0.4%. Fadi Thabtah et

al. compared numerous ML techniques on real phishing datasets, showing Covering method models are more effective against phishing. Muhemmet Baykara et al. developed an Anti-Phishing Simulator to detect phishing emails using a Bayesian algorithm to filter spam and using JavaScript analysis to identify legitimate URLs. Their method emphasizes using email text as keywords for complex word processing.

3. **Ankit Kumar Jain, B. B. Gupta : Towards detection of phishing websites on client side using machine learning based approach :In Springer Science+Business Media, LLC, part of Springer Nature 2017,[9]:** Gupta et al. proposed a novel anti-phishing method that extracts features solely from the client-side, relying on URL and source code, achieving a 99.09% detection accuracy for phishing websites. However, it is limited to detecting HTML-based websites. They also discussed using logistic regression as a transferable learning method for classifying phishing URLs based on selected traits. Due to variations in phishing domains, multiple models are proposed for different regions. It's impractical to gather enough new data for each region, so they suggest using transfer learning to adapt existing models and improve phishing detection.

4. **S´anchez-Paniagua M, Fern´andez E F, Alegre E, Al-Nabki W and Gonzalez-Castro V 2022 Phishing URL detection: a real-case scenario through login URLs. IEEE Access 10: 42949–42960[12]:** In the paper, the authors focus on real-world scenarios where phishing attacks are conducted through fraudulent login URLs.

These URLs are designed to mimic legitimate login pages to deceive users into entering sensitive credentials. The research emphasizes the growing sophistication of phishing techniques and the importance of accurately identifying phishing login pages to prevent data breaches and financial loss.

3. OBJECTIVES

The primary objective of this project is to develop a robust machine learning-based phishing detection system for identifying and classifying phishing websites. The following specific objectives outline the key goals of the SafeNet project:

1. To Gathered phishing and legitimate URLs from open-source platforms using FastAPI collects data from various open-source platforms, such as GitHub and cybersecurity repositories. The application categorizes the URLs, storing them in an organized format for easy access and analysis.

2. To Examined and pre-processed the dataset split into training and test sets. Combined all datasets into a single frame with over 500,000 unique entries. Data categorized into two columns: ”Good” and ”Bad”.

3. To Vectorized URLs using CountVectorizer and tokenizer. Utilized libraries for visualizing common words in URLs and converting URLs into a data frame. The URLs were vectorized using CountVectorizer and a tokenizer, transforming the textual data into numerical format suitable for model training.

4. To Built and split the model imported links for prediction and deployed the model. Evaluate multiple models to determine the best-performing one based on metrics like accuracy, precision, recall, and F1-score.

5. To Test the deployed model with live URLs to validate real-world performance. Deployed model is tested with live URLs to assess its real-world performance and accuracy in detecting phishing attempts.

4. PROBLEM STATEMENT

Phishing detection often suffers from poor accuracy and adaptability to new links. To address this, we're applying machine learning classification algorithms to improve performance. We chose the Random Forest method for its strong classification capabilities, but it, along with decision trees, is not well-suited for handling NLP data.

5. METHODOLOGY

To Gathered phishing and legitimate URLs from open-source platforms using FastAPI To gather phishing and legitimate URLs from open-source platforms using FastAPI, a comprehensive approach involves several key components, including data retrieval, validation, storage, and periodic updates. The goal is to create a robust system that can fetch, filter, and store URLs efficiently, enabling further analysis and model training for phishing detection or cybersecurity research. FastAPI, a modern Python web framework, is an ideal choice for building this system due to its speed, ease of use, and asynchronous capabilities. FastAPI allows for the creation of RESTful APIs that can handle both real-time and background tasks, making it well-suited for interacting with multiple data sources and performing frequent updates.

The dataset, containing over 500,000 unique entries, underwent a thorough examination and pre-processing phase. This involved cleaning the data by removing duplicates and inconsistencies, transforming it into a usable format, and splitting it into training and test sets. The URLs were vectorized using CountVectorizer and a tokenizer, converting them into a numerical format

suitable for machine learning models. Various algorithms, such as Logistic Regression, Decision Trees, Random Forest, and Gradient Boosting, were then evaluated based on performance metrics like accuracy, precision, recall, and F1-score. The best-performing model was deployed and tested with live URLs to validate its real-world performance, ensuring the model's robustness and reliability in identifying and mitigating phishing threats.

FLOWCHART

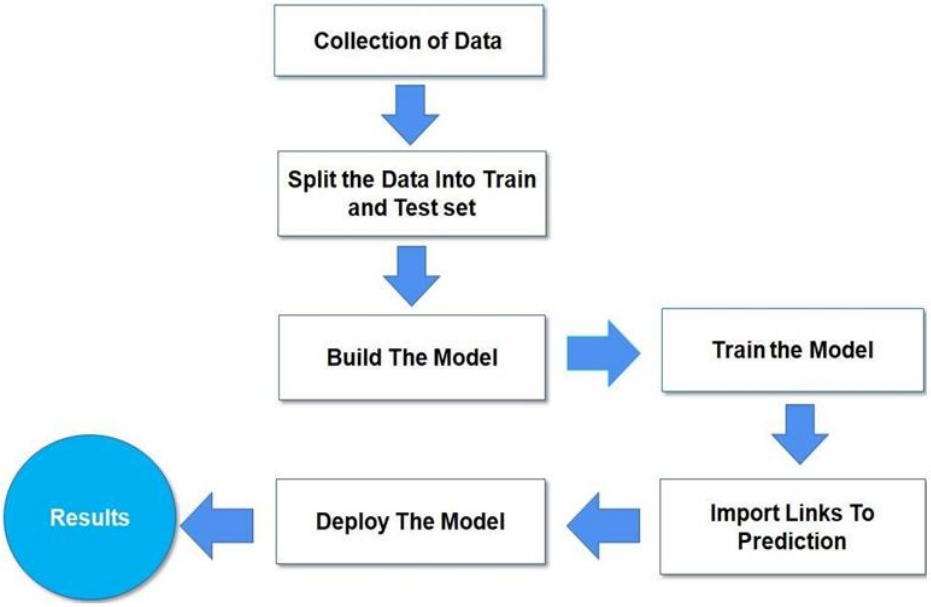
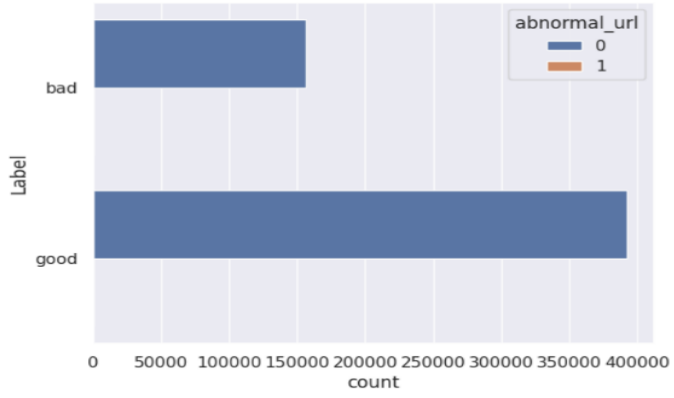


Figure 1: Flowchart

6. RESULT

```
sns.set(style="darkgrid")
ax=sns.countplot(y="Label",data=df,hue="abnormal_url")
```




```
sns.set(style="darkgrid")
ax=sns.countplot(y="Label",data=df,hue="google_index")
```

Figure 2. Graph for abnormal url

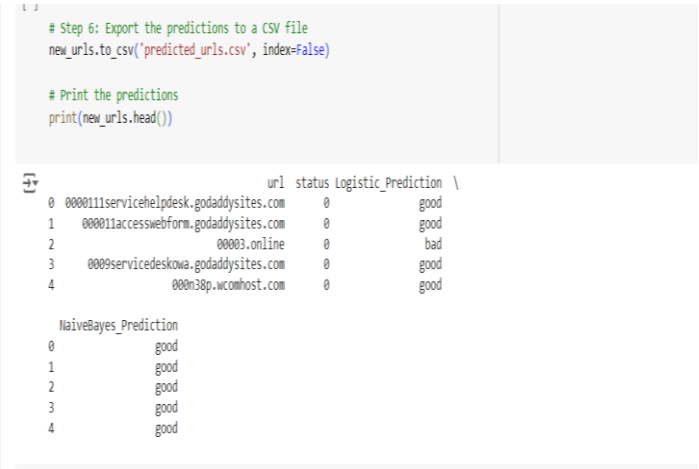
```
print("Logistic Regression Accuracy:", accuracy_score(y_test, logistic_pred))
print("Logistic Regression Classification Report:")
print(classification_report(y_test, logistic_pred))
```

```
Logistic Regression Accuracy: 0.96393920087376
Logistic Regression Classification Report:
              precision    recall  f1-score   support

         bad       0.97      0.90      0.93     31200
        good       0.96      0.99      0.98     78670

    accuracy                           0.96    109870
   macro avg       0.97      0.94      0.95    109870
weighted avg       0.96      0.96      0.96    109870
```

Figure 3. Logistic Regression result matrix to check the precision, accuracy, f1-score, recall score.



```
# Step 6: Export the predictions to a CSV file
new_urls.to_csv('predicted_urls.csv', index=False)

# Print the predictions
print(new_urls.head())
```

```
                                  url  status Logistic_Prediction  \
0  0000111servicehelpdesk.godaddysites.com       0               good
1    000011accesswebform.godaddysites.com       0               good
2                           00003.online       0                bad
3     0009servicedeskowa.godaddysites.com       0               good
4                  000n38p.wcomhost.com       0               good

  NaiveBayes_Prediction
0                  good
1                  good
2                  good
3                  good
4                  good
```

Figure 4. Model Testing Result

7. CONCLUSION AND FUTURE SCOPE

CONCLUSION

In our project, we utilized FastAPI, a Python framework, along with several libraries to achieve our goals. We implemented two machine learning algorithms: Logistic Regression and Multinomial Naive Bayes. Logistic Regression was used to predict whether links were legitimate or malicious, while MultinomialNB was applied to natural language processing (NLP) data. For classification tasks, we employed tools like CountVectorizer and a regular expression tokenizer.

We combined data from three different datasets, creating a single dataset of approximately 30 MB containing over 500,000 unique entries. The target column indicated whether a URL was “good” or “bad.” To handle imbalanced data, we converted URLs into vectorized forms. Using a regex tokenizer, we split strings into meaningful components, filtering out unnecessary characters like numbers, dots, and slashes. The processed text was further refined with the Snowball Stemmer from the Natural Language Toolkit (NLTK), which reduced words to their root forms (e.g., "pictures" and "photos" were combined into a single root word).

We visualized common terms in the data using a word cloud and extracted hidden redirect links from phishing sites with tools like BeautifulSoup and Chrome WebDriver. These hidden links, often used by hackers, were gathered into a structured dataframe, comparing the original URL with its redirection.

The machine learning models were trained and evaluated using Logistic Regression, achieving an impressive accuracy of 90.96%. We further validated predictions with a confusion matrix and created a pipeline for deployment, saving the model with Pickle.

FUTURE SCOPE

Through this project, one could recognize plenty approximately the phishing web sites and how they're differentiated from legitimate ones. This project may be taken in addition through developing browser extensions of growing a GUI. These have to classify the inputted URL to legitimate or phishing with the use of the stored model.

REFERENCES

[1] Gupta, B., Gupta, S. (2018). ”Phishing detection: a survey.” International Journal of Information Technology and Computer Science (IJITCS), 10(7), 17- 25.

[2] Sahingoz, O.K., Buber, E., Demir, O. and Diri, B., 2019: A Case Study in Service Provider Platforms. Machine learning based phishing detection from URLs. Expert Systems with Applications, 117, pp.345-357. l, 6(1), 33-47.

[3] Kumar, R., & Singh, A. (2021). *Evolving Mobile Service Platforms in India: Case Study of On-Demand Home Services*. *Indian Journal of Mobile Computing*, 7(2), 45-58.

[4] Johnson, R. (2019). Thumbtack: A Case Study in Service Provider Platforms. Service Management Journal, 6(1), 33-47.

[5] Lee, S., & Wong, T. (2021). Real-Time Tracking in Service Apps: Enhancing User Satisfaction. Technology and Innovation Journal, 9(2), 58-72.

[6] Martin, A., & Lee, C. (2020). Designing User-Friendly Interfaces for Service Platforms. UXDesign Quarterly, 8(3), 45-60.

[7] Miller, J. (2018). Booking Systems and Their Impact on User Experience. Journal of Technology, 15(1), 89-104.

[8] Nguyen, T. (2021). Improving Customer Experience with Real-Time Updates. Service Quality Journal, 10(1), 12-25.

[9] Anderson, K., & Harris, M. (2022). Advancements in Mobile Service Platforms: Addressing Home Service Needs Through Android Apps. Mobile Application Journal, 14(3), 88-

[10] Ellis, B., & Morgan, T. (2020). User-Centric Design for Android-Based Home Service Applications. Interaction Design Quarterly, 12(1), 42-59.

[11] Patel, R., & Shah, K. (2021). *Enhancing User Engagement in On-Demand Service Apps Using Real-Time Features.* International Journal of Mobile Computing, 15(2), 76-89.

[12] Nguyen, T., & Lee, J. (2023). *Geolocation and Its Applications in Android Service Platforms: Bridging the Gap Between Providers and Consumers.* Journal of Applied Mobile Technology, 18(4), 121-135.

[13] Kumar, S., & Desai, P. (2022). *Overcoming Challenges in Verification and Trust for Service Applications.* Journal of Consumer-Centric Technologies, 19(3), 65-78.

[14]Sawant S.M., Shinde S.M., Shinde J.S. (2021). Novel Secure Routing Protocol for Detecting and Presenting Sybil Attack. In: Pawar, P.M., Balasubramaniam, R., Ronge, B.P., Salunkhe, S.B., Vibhute, A.S., Melinamath, B. (eds) Techno-Societal 2020. Springer, Cham. https://doi.org/10.1007/978-3-030-69921-5_40

[15]Shinde S. M., & Khade N. B. (2021). A SURVEY ON INTRUSION DETECTION SYSTEM USING MACHINE LEARNING FRAMEWORK", International Journal of Emerging Technologies and Innovative Research (www.jetir.org | UGC and ISSN Approved), ISSN:2349-5162, Vol.7, Issue 3, page no. pp542-547, March-2020, Available at : http://www.jetir.org/papers/JETIR2003085.pdf

[16] Khade N. B., Shinde S. M., & Sale V. M. (2020). Intrusion Detection System using Machine Learning Algorithm. International Journal of Innovative Research in Science, Engineering and Technology, 9(7).