Detection of Phishing Sites Using Machine Learning

Pooja Maroti Jadhav[1], Tejashree Devendra More[2], Vaishnavi Hemant Bagul[3], Prof. S.V. Shardul[4]

**1,2,3 Department of Computer Engineering, Shatabdi Institute of Engineering and Research, Nashik, Maharashtra, India.*
**4 Professor, Department of Computer Engineering, Shatabdi Institute of Engineering and Research, Nashik, Maharashtra, India.*

Abstract - The content of phishing websites and online-based material provides a variety of indicators. One of the criminal attacks that are successful in the online world is phishing sites that direct users to a phishing website that looks and acts like a legitimate website in order to steal the victim's personal and sensitive information. The Extreme Learning Machine-based version that was suggested was excellent at spotting phishing websites. Internet page types vary greatly in terms of their characteristics. So, in order to protect against any phishing attack, we must use a set of web page features. To defend against these threats, a machine learning strategy is used. The phishing dataset that is planned to be imported, authentic URLs from the database, and also the data that is collected are all pre-processed. Four groups of URL characteristics are used to detect phishing websites: domain, address, anomalous based, HTML, and JavaScript features. Data that has been analyzed is used to extract URL characteristics and produce URL attribute values. ML approaches are used to analyze URLs and determine the threshold value and range value for URL properties. This project's goal is to develop an ELM categorization for a number of database characteristics and a few phishing sites.

Keywords— *Support Vector Machine (SVM), Extreme Learning Machine (ELM),URL Phishing Websites, Browser add-ons, Random Forest*

1. INTRODUCTION

Phishing is the tactic of pretending to be a trustworthy entity during an internet encounter in order to get sensitive information such as usernames, passwords, and credit card numbers, usually for malicious purposes [1]. Nowadays, safety researchers are concerned about the idea of phishing since it's simple to create a fake website that mimics a genuine website. Although experts can spot fake websites with ease, most people struggle to tell them apart, and as a result, they become targets of phishing assaults. The attacker's primary goal is to steal login information for financial institution accounts. Customers who fall victim to phishing cost American businesses $ 2 billion annually[3].

According to the 3rd Microsoft Computing Safety Index Report, which was released in February 2014, the yearly global effect of phishing may be close to or greater than $5 billion. Lack of human apprehension is one of the reasons these attacks are successful. It is extremely difficult to combat phishing attacks since they exploit users' susceptible data, yet it is crucial to change phishing detection tactics [3].

In this attack, the phisher creates a fake internet webpage by copying the content of the legitimate website, making it impossible for a user to distinguish between legitimate and phishing websites. Social engineering tactics take advantage of gullible victims by tricking them into thinking they are dealing with a reliable, legitimate entity by utilizing fictitious email accounts and email messages[1].

The general method of identifying phishing websites by adding blacklisted URLs and IP addresses to the antivirus database is sometimes referred to as the "blacklist" method. In order to avoid blacklists, attackers employ cunning tactics to trick users into altering the URL to appear authentic using obfuscation and a variety of straightforward approaches, such as fast-flux and automatically produced proxies to host a website[3].

It is possible to use ML to become familiar with and create excellent data outputs. The system uses the use-case of phishing website detection to demonstrate this notion.

Spear phishing and email phishing scams are two prominent ways that phishing assaults are carried out, thus clients must be aware of the outcomes and can no longer place their whole reliance on any unofficial protection software. Using ML, the drawback of the current technique may be fixed[3].

Using implicit programming, this branch of artificial intelligence has the capacity to learn. Unsupervised learning, reinforcement learning, and other machine learning methods are all supervised. The many machine learning methods include:

A. Supervised learning
B. Unsupervised learning
C. Reinforcement learning

2. PRIOR WORK

Before the development of "PhishGuard Pro," several approaches to phishing detection existed in the cybersecurity domain. Traditional methods relied

heavily on static blacklists of known phishing URLs and heuristics to identify suspicious websites based on characteristics like URL structure or webpage content. While these methods provided some level of protection, they often struggled to keep pace with the dynamic and evolving nature of phishing attacks. Additionally, machine learning models were occasionally employed to enhance detection accuracy, but these approaches were limited by the availability of labeled datasets and the complexity of feature engineering.

Moreover, existing systems typically lacked seamless integration with web browsers, making them less accessible and user-friendly for individuals seeking reliable protection against phishing threats. User education efforts were also fragmented, leaving many users vulnerable to social engineering tactics employed by attackers. Recognizing these limitations, the development of "PhishGuard Pro" aimed to address these challenges comprehensively.

By leveraging advanced machine learning algorithms, heuristic analysis, and seamless integration as a browser extension, "PhishGuard Pro" represents a significant advancement in phishing detection technology. It offers real-time detection and prevention of phishing attempts, empowering users with visual indicators and educational resources to recognize and avoid malicious websites effectively. Furthermore, with a strong emphasis on user privacy and continuous updates from external databases of known phishing sites, "PhishGuard Pro" sets a new standard for online security solutions, promising a safer and more secure browsing experience for users worldwide.

3.METHODOLOGY

Certainly, let's outline the methodology for developing "PhishGuard Pro" using the provided steps:

1**. Define Objectives and Scope:**

- Clearly define the objectives of the project, focusing on enhancing online security through effective phishing detection.
- Outline the scope of the project, including the features and functionalities to be included in "PhishGuard Pro."

2**. Data Collection:**

- Gather a diverse dataset containing both legitimate and phishing website samples for training and testing the machine learning models.
- Ensure the dataset is representative of real-world phishing threats and includes a variety of features relevant to phishing detection.

3. Preprocessing:

- Clean the dataset by handling missing values, removing duplicates, and addressing any inconsistencies.
- Perform feature engineering to extract relevant features and transform them into a suitable format for analysis.

4. Summarization:

- Summarize the dataset to gain insights into its characteristics, distribution of features, and potential correlations.
- Conduct exploratory data analysis to identify patterns and trends that may aid in the development of the phishing detection system.

5**. Synthesis:**

- Develop machine learning models, heuristic analysis techniques, and other algorithms for phishing detection based on the dataset and objectives of the project.
- Implement real-time detection mechanisms and integrate them into the browser extension interface.

6. Evaluation:

- Evaluate the performance of the developed models using appropriate metrics such as accuracy, precision, recall, and F1-score.
- Conduct cross-validation and testing on separate datasets to assess the generalization capability of the models.

7**. Integration:**

- Integrate the developed algorithms, user interface components, and backend systems to create a cohesive "PhishGuard Pro" system.
- Ensure seamless communication and interaction between different modules of the system.

8**. Deployment and Maintenance:**

- Deploy "PhishGuard Pro" as a browser extension, making it accessible to users across different platforms and web browsers.
- Establish a maintenance plan for regular updates, bug fixes, and enhancements to keep the system up-to-date and effective against evolving phishing threats.

9**. Ethical Considerations:**

- Address ethical considerations related to user privacy, data protection, and transparency in the system's operations.
- Ensure compliance with relevant regulations and standards governing online security and privacy.

10. User Feedback and Iteration:

- Incorporate mechanisms for collecting user feedback on the effectiveness and usability of "PhishGuard Pro."
- Use user feedback to iterate on the system, making improvements and adjustments based on user preferences and experiences.

3. LITERATURE SURVEY

The system makes use of supervised learning, a type of machine learning. The Random Forest approach was chosen in this instance for its overall performance in smart ranking. The goal is to identify the ideal combination to train the classifier by analyzing the attributes of phishing websites and tracking down the best performing classifier. Hence, the paper's accuracy is 98.8% and there are 26 [9] different characteristics utilized.

To increase the accuracy of phishing detection, a variety of data mining approaches are used. A feature selection methodology is also used to increase the precision of the classification model by removing the most effective features and determining the best outcome.

The fast ML framework Vowpal Wabbit is used for feature hashing, which uses hash functions to hash feature words in n memory indexes. In order to differentiate any unsolicited approaches, the study offers 2-class logistic regression, boosted decision trees, neural networks, and SVM[2].

A technological real-time method is anticipated with the aim of successfully shielding a customer from client-side phishing assaults. Hyperlinks on a webpage are the only useful component that may be utilized to identify assaults. To identify a DNS incursion of devices, Google public DNS is checked with the IP address of questionable websites[4].

This study describes methods for detecting phishing websites by comparing a wide range of legitimate and malicious URLs using machine learning techniques. Various techniques that support lexical characteristics, host, and page consequentiality attributes are used to detect phishing websites. To encourage a deeper knowledge of the structure of URLs that leads to assault, several data processing methods are examined for feature analysis. The best ML algorithmic strategy for distinguishing between malicious and benign websites may be chosen with the use of finely tailored parameters[5].

The AUP lifecycle is suggested by the study as a way to shorten the development phase. The admin has the power to separate URLs into banned and whitelisted categories; once these websites are included, he has the ability to update, amend, and remove them. Different color backgrounds are used to classify phished or blacklisted Websites for the benefit of users. When the link is clicked, the non-blacklisted Website opens.[6]

To deal with the complexity of monitoring requirements for every modern scenario, the suggested computer finds and chooses the appropriate options. This programme focuses on the traits that show up during the monitoring phase and on the broad scope of its capability[7].

The technique employs a machine learning (ML) classifier and agent-predicted architecture to counter various phishing assaults. The distributed internet needs the use of several agents that communicate using peer-to-peer technology. For identifying and addressing web-based phishing attempts, the study offers a tiered multiple-agent approach. Multi-agents' duties include extracting URLs, seeing scripts and phished URLs, and blocking. The method works well for understanding and studying in line with environmental changes [8].

The research proposes a deep learning model for phishing detection that mainly relies on 1D CNN. The algorithm examines a typical dataset that contains 6,157 examples of legal websites and 4,898 instances of phishing sites. This model significantly outperforms other popular ML classifiers that have been assessed using the same dataset. The final findings show that, in comparison to other models, the CNNbased strategy provides the highest accurate result and also finds new phishing websites[9].

4.PROCESS OF PROJECT

Certainly, let's outline the process of the project, "PhishGuard Pro," using the provided steps:

1. **Requirements Gathering:**

- Gather requirements from stakeholders, including end-users, cybersecurity experts, and developers, to understand the desired features and functionalities of "PhishGuard Pro."
- Define the scope, objectives, and success criteria for the project based on the gathered requirements.

2**. Transcription:**

- Transcribe the gathered requirements into a detailed project plan, outlining the tasks, timelines, and resources required for each stage of development.
- Ensure clear communication and alignment between stakeholders to avoid misunderstandings and scope creep.

3. Summarization:

- Summarize the project plan and requirements into actionable tasks, breaking down the development process into manageable components.
- Prioritize tasks based on importance, dependencies, and project timelines to ensure efficient progress.

4. **Integration:**
   - Integrate various components of the system, including machine learning algorithms, user interface elements, backend systems, and external databases.
   - Ensure seamless communication and interoperability between different modules of "PhishGuard Pro" for a cohesive user experience.

5. User Interface:
   - Design and develop the user interface for "PhishGuard Pro," focusing on usability, intuitiveness, and visual indicators for phishing detection.
   - Incorporate user feedback to refine the interface and improve user engagement.

6. Testing and Refinement:
   - Conduct thorough testing of "PhishGuard Pro" to identify and address any bugs, errors, or usability issues.
   - Perform functional testing, usability testing, and security testing to ensure the system meets quality standards and user expectations.
   - Iterate on the design and functionality based on testing results and user feedback, refining the system for optimal performance.

7. Deployment:
   - Deploy "PhishGuard Pro" as a browser extension, making it available to users through popular web browsers such as Google Chrome, Mozilla Firefox, and Microsoft Edge.
   - Ensure smooth deployment and user onboarding processes, providing resources and guidance for installing and using the extension.

8. Maintenance and Updates:
   - Establish a maintenance plan for "PhishGuard Pro," including regular updates, bug fixes, and enhancements to address evolving threats and user needs.
   - Monitor system performance and user feedback post-deployment, incorporating updates and improvements as necessary to ensure ongoing effectiveness and usability.
   - This process outlines the systematic approach to developing and deploying "PhishGuard Pro," ensuring that all stages of the project are executed efficiently and effectively to deliver a high-quality and user-friendly phishing detection solution.

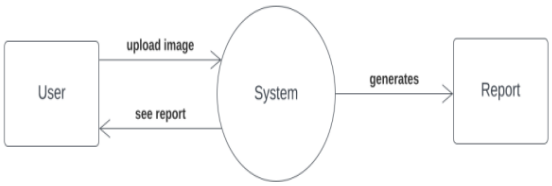
5. IMPLEMENTATION

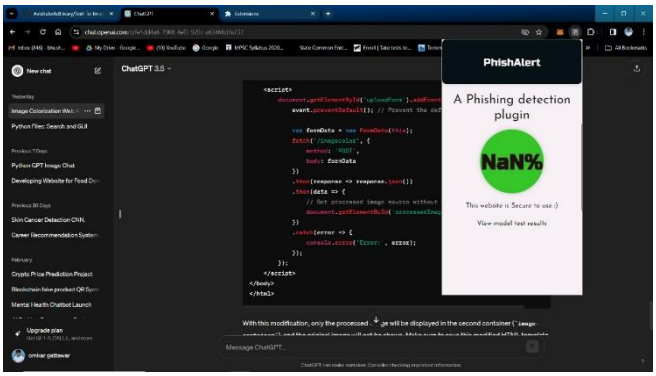
The data flow of the "Skin Cancer Detection using Python" system is a complex yet meticulously orchestrated process that encompasses various stages, from user interaction to machine learning model predictions. The journey begins with users accessing the web-based interface, where they can upload skin lesion images for analysis. Once uploaded, these images initiate a series of data flow processes. The system preprocesses the images, standardizing them for consistency in format and resolution.

The preprocessed images then enter the machine learning pipeline, where the Convolutional Neural Networks (CNNs) come into play. These pre-trained models have undergone rigorous training on a diverse dataset of skin lesion images, enabling them to recognize intricate patterns and features indicative of different skin conditions, including potential cancerous lesions. The machine learning models analyze the input images and generate diagnostic assessments based on learned patterns.

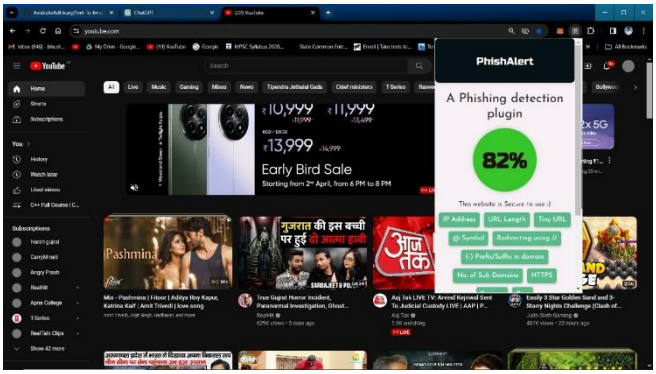
Simultaneously, user data, including uploaded images and diagnostic results, is securely stored in a database. This database serves as a repository for user information, ensuring the continuity of the user's journey within the system and providing a historical record for future reference.

The diagnostic results, obtained from the machine learning models, are then communicated back to the user through the web-based interface. Users can view these results, which include information about the likelihood of a skin lesion being cancerous and any recommended actions. Additionally, users have access to educational resources embedded in the platform, providing insights into skin health, risk factors, and preventive measures.





The system also facilitates communication between users and healthcare professionals. In cases where further evaluation is recommended, users can engage in remote consultations, leveraging the collaborative interfaces within the platform. This interaction supports a comprehensive approach to skin health, allowing users to seek expert advice and facilitating timely medical interventions.



The entire data flow is designed with privacy and security as top priorities. User data is encrypted during transmission and securely stored, adhering to data protection regulations. The system's commitment to ethical considerations is embedded in every step of the data flow, with continuous monitoring and mitigation strategies in place to address biases and ensure fair and transparent diagnostic assessments.

In summary, the data flow of the "Skin Cancer Detection using Python" system is a dynamic and interconnected process, seamlessly weaving user interactions, machine learning analysis, data storage, and educational components. This intricate flow is meticulously designed to provide users with accurate diagnostic assessments while prioritizing privacy, security, and ethical considerations throughout the entire journey within the system.

CONCLUSION

Websites can be used to establish a system for everything from record entry to scientific applications. The statistics used as input may be processed, and the output of the processing may be information. Websites are now used in a wide range of disciplines, including medicine, technology, business, education, and economics. Due to its widespread use, hackers may also utilize it as a tool for evil ends. A rogue object makes an apparent phishing attack. Many research contributions provide completely original methods, approaches for locating phishing URLs, and applications of those methodologies. The equipment's goal is to create a category for the classification of phishing as one of the attack types that cyber threats use. The system warns the user of phishing URLs by recommending safe URLs even before they are discovered on such websites, ultimately preventing a phishing attack. The intensive learning tool will be used as a result. We're going to utilize a dataset from the UCI for the purposes of this examination.

REFERENCES

[1] Oza Pranali P, Deepak Upadhyay, Review on Phishing Sites Detection Techniques, IJERT, ISSN: 2278-0181, 04,April-2020

[2] Meenu, Sunila Godara, Phishing Detection using Machine Learning Techniques, IJEAT, ISSN: 2249 – 8958, 2, December,2019.

[3] Sandeep Kumar Satapathy, Shruti Mishra, Pradeep Kumar Mallick, LavanyaBadiginchala, Ravali Reddy Gudur, SiriChandanaGuttha, IJITEE, ISSN: 2278- 3075, June2019.

Ankit Kumar Jain and B.B. Gupta EURASIP Journal On Information Security (2016)2016:9

[4] Joby James, Sandhya L, Ciza Thomas, Detection of Phishing URLs Using Machine Learning Techniques, 2013 International Conference on Control Communication and Computing (ICCC), December 2013.

[5] Mohammed Hazim Alkawaz, Stephanie Joanne Steven, AsifIqbalHajamydeen, Detecting Phishing Websites Using Machine Learning, 2020 16th IEEE International Colloquium on Signal Processing & its Applications, 28-29 Feb. 2020.

[6] Suleiman Y. Yerima, Mohammed K. Alzaylaee, High Accuracy Phishing Detection Based on Convolutional Neural Networks, IEEEXplore.

[7] Megha N, KR Ramesh Babu, Elizabeth Sherly, An Intelligent System for Phishing Attack Detection and Prevention, IEEE Xplore ISBN: 978-1-7281-1261-9, 2019 IEEE.

[8] AmaniAlswailem, BashayrAlabdullah, Norah Alrumayh, Dr. Aram Alsedrani, Detecting Phishing Websites UsingMachine Learning 978-1-7281-0108-8/19/ 2019 IEEE.