A Project report on

PHISHING DETECTION

A Dissertation submitted to JNTU Hyderabad in partial fulfillment of the academic requirements for the award of the degree.

Bachelor of Technology

in

Computer Science and Engineering (Cyber Security)

Submitted by

G.CHARAN SAI

(20H51A6279)

A.RISHIKA

(20H51A6297)

G.DHEERAJ REDDY

(20H51A62A0)

Under the esteemed guidance of

Dr. Punyaban Patel

(Professor)

Department of Computer Science and Engineering (Cyber Security)

CMR COLLEGE OF ENGINEERING & TECHNOLOGY

(UGC Autonomous)

*Approved by AICTE *Affiliated to JNTUH *NAAC Accredited with A+ Grade

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD - 501401.

2020 - 2024

CMR COLLEGE OF ENGINEERING & TECHNOLOGY

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD – 501401

Department of Computer Science and Engineering (Cyber Security)

CERTIFICATE

This is to certify that the Major Project report entitled **"Phishing Detection"** being submitted by Garikapati Charan Sai (20H51A6279), Akula Rishika (20H51A6297), Garlapati Dheeraj Reddy (20H51A62A0) in partial fulfillment for the award of **Bachelor of Technology in Computer Science and Engineering (Cyber Security)** is a record of bonafide work carried out his/her under my guidance and supervision.

The results embodies in this project report have not been submitted to any other University or Institute for the award of any Degree.

Dr. Punyaban Patel
Professor
Dept. of CSE (Cyber Security)

Dr. R. Venkateswara Reddy
Associate Professor and HOD
Dept. of CSE (Cyber Security)

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

With great pleasure we want to take this opportunity to express my heartfelt gratitude to all the people who helped in making this project work a grand success.

We are grateful to **Dr. Punyaban Patel, Professor** , Department of **Computer Science and Engineering (Cyber Security)** for his valuable technical suggestions and guidance during the execution of this project work.

We would like to thank **Dr. R. Venkateswara Reddy,** Head of the Department of **Computer Science and Engineering (Cyber Security)**, CMR College of Engineering and Technology, who is the major driving forces to complete my project work successfully.

We are very grateful to **Dr. Ghanta Devadasu**, Dean-Academics, CMR College of Engineering and Technology, for his constant support and motivation in carrying out the project work successfully.

We are highly indebted to **Major Dr. V. A. Narayana,** Principal, CMR College of Engineering and Technology, for giving permission to carry out this project in a successful and fruitful way.

We would like to thank the **Teaching & Non- teaching** staff of Department of CSE (Cyber Security) for their co-operation.

We express our sincere thanks to **Shri. Ch. Gopal Reddy**, Secretary, CMR Group of Institutions, and **Shri. Ch. Abhinav Reddy**, CEO, CMR Group of Institutions for their continuous care and support.

Finally, We extend thanks to our parents who stood behind us at different stages of this Project. We sincerely acknowledge and thank all those who gave support directly and indirectly in completion of this project work.

Garikapati Charan Sai (20H51A6279)
Akula Rishika (20H51A6297)
Garlapati Dheeraj Reddy (20H51A62A0)

DECLARATION

We hereby declare that results embodied in this Report of Project on "**PHISHING DETECTION**" are from work carried out by using partial fulfillment of the requirements for the award of B. Tech degree. We have not submitted this report to any other university/institute for the award of any other degree.

NAME	**ROLL NO.**	**SIGNATURE**
G. Charan Sai	20H51A6279	
A. Rishika	20H51A6297	
G. Dheeraj Reddy	20H51A62A0	

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

Phishing detection is the process of identifying phishing attacks in their early stages, warning users and administrators, and ideally, mitigating the threat. Phishing is a type of cyberattack where attackers attempt to trick individuals into revealing sensitive information, such as login credentials or financial details, by impersonating a trustworthy entity via electronic communication, typically email. Identifying these phishing websites is typically a challenging task because phishing is mainly a semantics-based attack, that mainly focuses on human vulnerabilities. To address this issue, we proposed the Phishing Detection project.

Our project is designed to enhance online security by detecting and promptly alerting users to potential phishing and malware threats. This comprehensive approach involves URL analysis, Database comparison, Greek Alphabet Analysis, Homographic attack detection and Port forwarding detection using machine learning to assess the legitimacy of URL's.

CHAPTER 1
INTRODUCTION

CHAPTER 1
INTRODUCTION

In today's rapidly advancing digital landscape, the prevalence of online communication and transactions has ushered in a new era of convenience and connectivity. However, this digital evolution has also brought about a significant rise in cyber threats, with phishing attacks emerging as a prominent concern. Phishing, a deceptive practice wherein cybercriminals employ cunning tactics to deceive users into divulging sensitive information, poses a substantial risk to both individuals and organizations alike. Such attacks threaten the confidentiality and integrity of personal and organizational data, potentially resulting in financial loss, identity theft, and reputational damage. As malicious actors continue to refine their techniques, traditional security measures often prove inadequate in effectively thwarting these evolving threats. Consequently, there is an urgent and pressing need for innovative solutions capable of detecting and neutralizing phishing attempts in real-time.

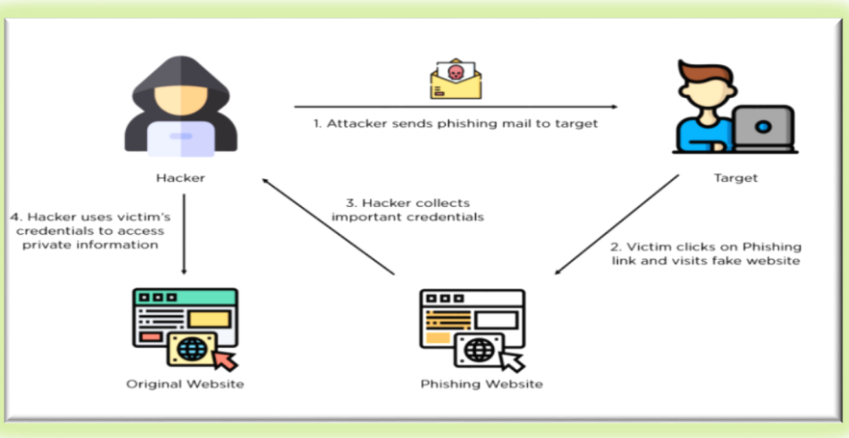

Fig.1: Phishing Process

Fig.2: Phishing Website

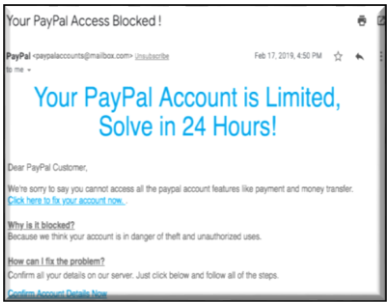

Fig.3: Phishing Email

1.1 Problem Statement

Our proposed system is designed to enhance online security by detecting and promptly alerting users to potential phishing and malware threats. This comprehensive approach involves URL analysis, Database comparison, Greek Alphabet Analysis, Homographic attack detection and Port forwarding detection using machine learning.

1.2 Research Objective

The research objective of our proposed solution is to advance the field of online security through the development and evaluation of a comprehensive phishing and malware detection system. This system is designed to leverage machine learning techniques for the detection and prompt alerting of users to potential threats. The key research objectives are as follows:

a) ***Algorithm Development***: Investigate and develop machine learning algorithms tailored for phishing and malware detection, focusing on enhancing accuracy and efficiency in identifying malicious URLs.

b) ***Feature Analysis and Selection***: Explore and analyze features derived from URL analysis, database comparison, Greek alphabet analysis, homographic attack detection, and port forwarding detection. Evaluate the relevance and effectiveness of these features in improving detection performance.

c) ***Data Collection and Preparation***: Collect diverse and representative datasets encompassing legitimate URLs, phishing attempts, and malware-infected content. Implement rigorous data preprocessing and augmentation techniques to ensure the quality and diversity of the training data.

d) ***Model Training and Evaluation***: Train and evaluate machine learning models using the collected datasets and selected features. Assess the performance of the models through rigorous evaluation metrics such as accuracy, precision, recall, and F1-score.

e) ***Real-time Detection and Alerting***: Investigate real-time detection mechanisms to promptly identify and alert users to potential phishing and malware threats. Develop strategies for efficient processing of incoming data streams and timely delivery of alerts to end-users.

1.3 Literature Review

Three techniques were proposed for categorization from the writers of the paper [8], Rishikesh Mahajan and Irfan Siddavatam: Decision Tree, Random Forest, and Support Vector Machine. Their sample included 19,653 phishing URLs and 17,058 benign URLs, each with 16 attributes, collected from the websites of Alexa and PhishTank, respectively. There were two sections in the data sets: identifying and training in the following ratios: 50:50, 70:30, and 90:10, respectively. The rates of false positive and negative and accuracy score were among the metrics used to assess performance. They achieved 97.14% accuracy using the Random Forest algorithm as it has low false negative rate. The study found that increasing the quantity if the training data gest better accuracy.

Jitendra Kumar et al.'s study from [9] trained a variety of classifiers, including K Nearest Neighbor, Random Forest, Naive Bayes Classifier, Logistic Regression, and Decision Trees, using variables extracted from the lexical structure of the URL. The URL dataset was created with volatility, overfitting, biased training, and unbalanced data in mind. The dataset had an equal number of tagged phishing and genuine URLs and was further split into testing and training groups using a 7:3 ratio. Even though the AUC values among all the categorization were almost equal, the Naive Bayees Classifier turned out to be more suitable because it had the highest value. The most accuracy that Naive Bayes could attain was 98%, with precision=1, recall=0.95, and F1-score= 0.97. Mehmet Korkmaz et al. developed a detection of phishing scams by ML in [10] by utilising eight different algorithms on three different datasets.

The techniques used were XGBoost, Random Forest (RF), Naive Bayes (NB), K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Logistic Regression (LR), K-Nearest Neighbor (KNN), and Artificial Neural Network (ANN). It was discovered that the models using LR, SVM, and NB had low accuracy rates. The techniques used were XGBoost, Random Forest (RF), Naive Bayes (NB), K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Logistic Regression (LR), K-Nearest Neighbor (KNN), and Artificial Neural Network (ANN). It was discovered that the models using LR, SVM, and NB had low accuracy rates.

The 32-feature Kaggle dataset underwent principal component analysis (PCA) and various feature selection procedures. The dataset benefits from feature selection since it eliminates unnecessary or redundant data. The proposed model used the REF, Relief-F, IG, and GR algorithms for feature selection before turning to PCA. Random Forest achieved an accuracy rate of 97%.It handled the over-fitting issue better and was less erratic. Using the UCI dataset, Abdulhamit Subasi et al. launched an intelligent phishing detection software in [12]. To recognize phishing websites, a numerous machine learning strategies were used among the classifiers are Artificial Neural Networks (ANN), K-Nearest Neighbours (K-NN), Support Vector Machines (SVM), C4.5 Decision Trees, Random Forests (RF), and Rotation Forests (RoF).

CHAPTER 2
BACKGROUND WORK

CHAPTER 2
BACKGROUND WORK

2.1 PhishTank

2.1.1. Introduction

PhishTank is a website and web service that provides information about phishing sites which is shown in Fig.4 . It offers a real-time dataset of phishing websites. PhishTank is not a piece of software and doesn't run on your computer. PhishTank is commonly used in academic studies. Preliminary experiments show that PhishTank can successfully identify 91.44% of phishing targets.

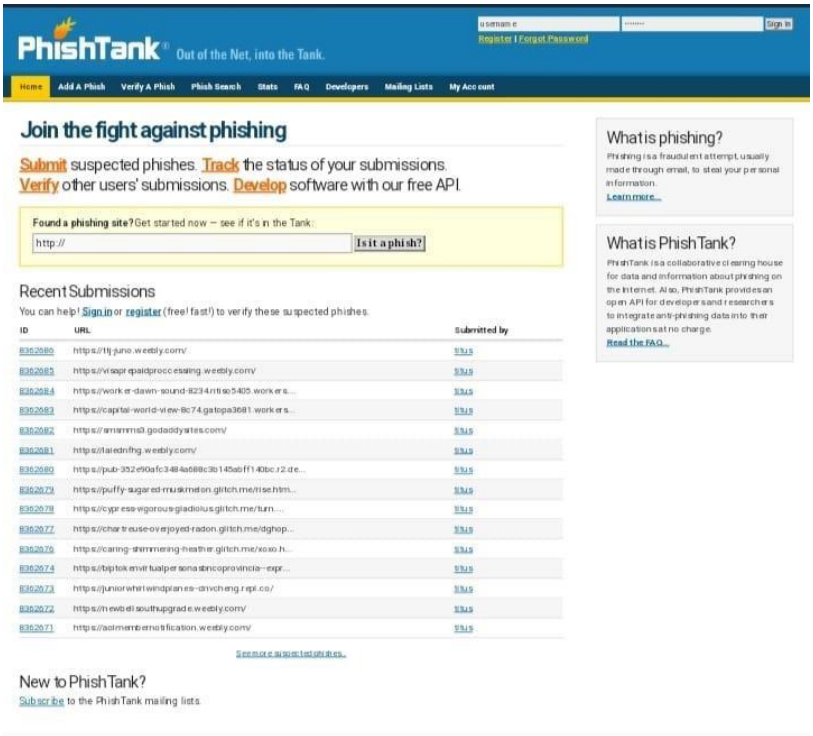

Fig.4: PhishTank Website

2.1.2. Merits, Demerits and Challenges

Merits of PhishTank Website

a) ***Community-driven Platform*****:** PhishTank operates as a collaborative platform where users can report and verify phishing websites, fostering a sense of collective responsibility in combating phishing attacks.

b) ***Timely Detection and Response*****:** By crowdsourcing reports and verifications from users, PhishTank can quickly detect and respond to emerging phishing threats, helping to mitigate potential damage before it escalates.

c) ***Centralized Repository of Phishing Data*****:** PhishTank serves as a centralized repository of phishing data, providing researchers, security professionals, and organizations with valuable insights into the tactics and trends of phishing attacks.

Demerits of PhishTank Website

a) ***Potential for False Positives*****:** Since PhishTank relies on user submissions and verifications, there is a risk of false positives where legitimate websites are incorrectly flagged as phishing sites, potentially leading to unnecessary disruptions or false accusations.

b) ***Limited Resources for Verification*****:** PhishTank's effectiveness depends on the availability of resources for verifying reported phishing URLs. If there are insufficient moderators or automated verification mechanisms, some phishing sites may go undetected or unverified.

c) ***Susceptibility to Manipulation*****:** Like any crowdsourced platform, PhishTank may be susceptible to manipulation or abuse by malicious actors who attempt to influence or undermine the integrity of the verification process.

Challenges of PhishTank Website

a) ***Keeping Pace with Evolving Threats*:** Phishing tactics and techniques are constantly evolving, presenting a challenge for PhishTank to stay ahead of emerging threats and adapt its detection mechanisms accordingly.

b) ***Maintaining Data Integrity*:** Ensuring the accuracy and integrity of the data in PhishTank's repository requires ongoing efforts to verify reported URLs and prevent the spread of false information.

c) ***Scaling Verification Efforts*:** As the volume of reported phishing URLs increases, PhishTank may face challenges in scaling its verification efforts to keep pace with the growing demand for timely detection and response.

d) ***Addressing Global Diversity*:** Phishing attacks are global in nature, and PhishTank must address the diversity of languages, cultures, and regions to effectively combat phishing threats on a global scale.

e) ***Balancing Automation and Human Oversight*:** Finding the right balance between automated verification processes and human oversight is essential for maintaining the accuracy and reliability of PhishTank's detection efforts while efficiently handling large volumes of reported URLs.

2.2 URLVoid

2.2.1. Introduction

URLVoid was a website that provided a free online tool for website reputation and safety analysis which is shown in Fig.5 . Users could enter a website URL, and URLVoid wouldcheck the site against multiple databases to provide information about its safety, potential threats, and reputation. The service was useful for individuals and organizations to assess the security of a website before visiting it.

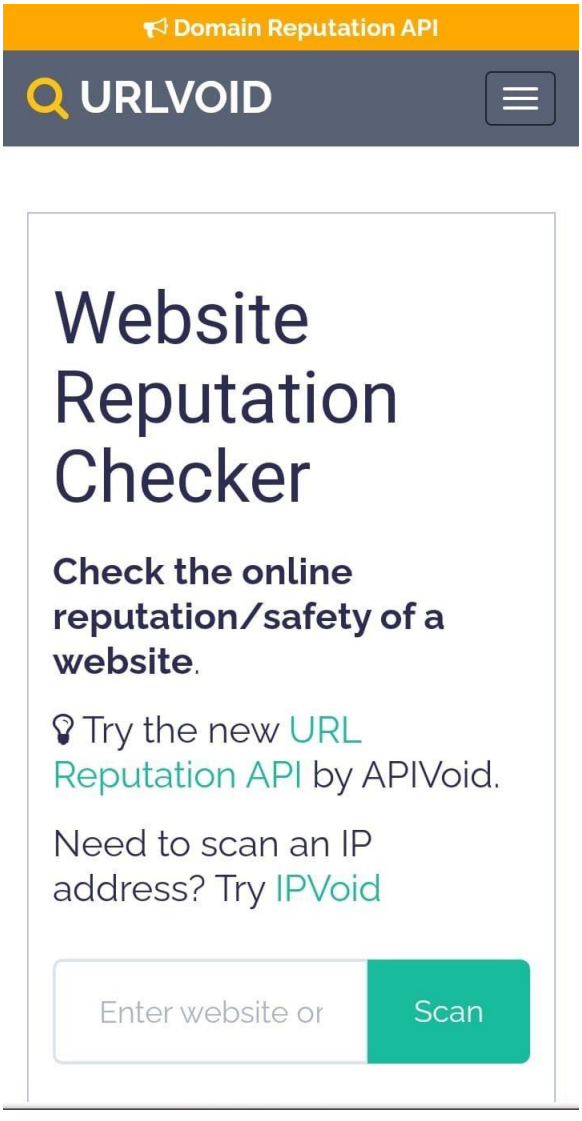

Fig.5: URL Void Website

2.2.2. Merits, Demerits and Challenges

Merits of URLVoid Website

a) ***User-Friendly Interface*****:** URLVoid offers a simple and intuitive interface that allows users to quickly and easily check the reputation of URLs. This accessibility makes it convenient for users of all levels of technical expertise to utilize the service effectively.

b) ***Real-Time Updates*****:** URLVoid provides real-time updates on the reputation of URLs, ensuring that users have access to the latest information on potential security threats. This timely information allows users to make informed decisions and take appropriate action to protect themselves against malicious websites.

c) ***Free and Accessible*****:** URLVoid is available as a free service, making it accessible to users without any cost barriers.

Demerits of URLVoid Website

a) ***Limited Coverage***: URLVoid's analysis may be limited by the scope of the security services and databases it integrates with. As a result, there may be instances where certain URLs are not adequately covered, leading to potential gaps in the analysis.

b) ***Reliance on External Data Sources***: URLVoid relies on external security services and databases for its analysis, which means its accuracy and reliability are contingent on the quality and timeliness of the data provided by these sources.

c) ***Potential False Positives and Negatives*****:** Like any automated security tool, URLVoid may generate false positives (incorrectly flagging legitimate URLs as malicious) or false negatives (failing to identify malicious URLs).

Challenges of URLVoid Website

a) ***Keeping Pace with Emerging Threats*:** URLVoid faces the ongoing challenge of keeping pace with emerging cyber threats and evolving attack techniques. As cybercriminals develop new methods to evade detection, URLVoid must continually update its analysis methods and data sources to effectively identify and mitigate these threats.

b) ***Ensuring Data Privacy and Security*:** URLVoid must prioritize data privacy and security to protect the sensitive information of its users. This includes implementing robust security measures to safeguard user data and adhering to relevant privacy regulations and standards.

c) ***Maintaining Service Reliability*:** URLVoid must ensure the reliability and availability of its service to meet the needs of users. This includes maintaining uptime, minimizing latency, and addressing any technical issues or disruptions promptly to ensure a seamless user experience.

2.3 CheckPhish

2.3.1. Introduction

CheckPhish is a free, AI-powered scanner that detects phishing and fraudulent URLs shown in Fig.6 . It uses machine learning to detect digital threats, including look-alike domains and fake sites. CheckPhish can be used right from your inbox. CheckPhish is a community offering from Bolster. It's used by some of the largest companies in the world, including nearly every Fortune 500 company.

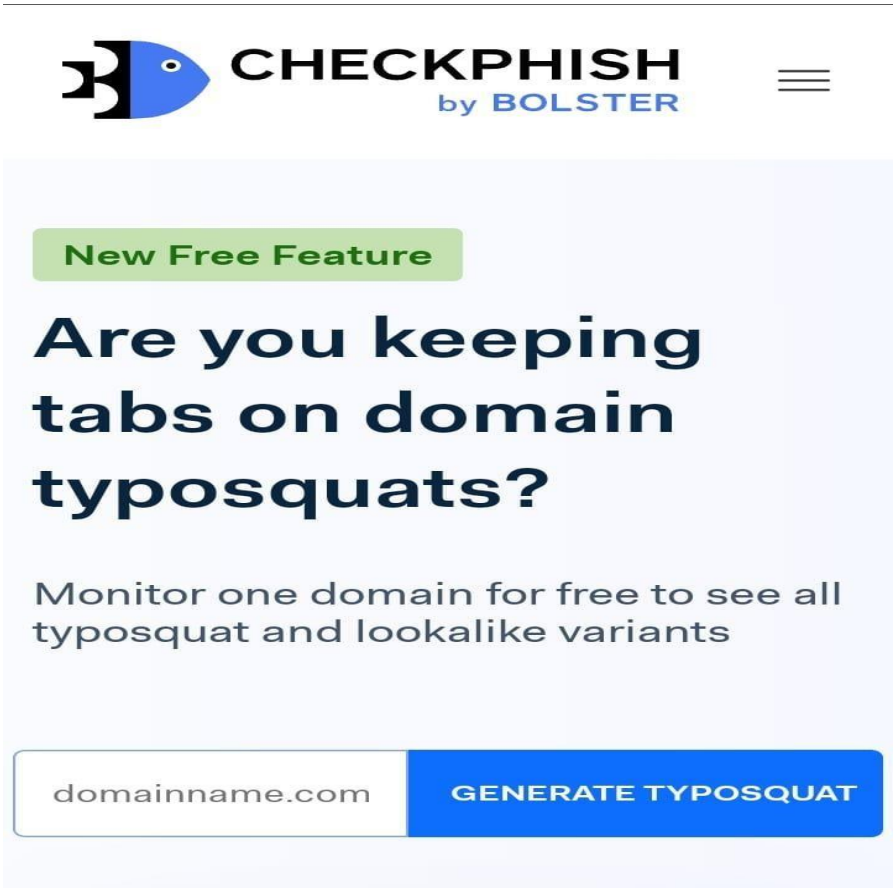

Fig.6: CheckPhish Website

2.3.2. Merits, Demerits and Challenges

Merits of CheckPhish Website

a) ***Improved Security*:** CheckPhish provides users with a tool to verify the legitimacy of URLs, helping them avoid falling victim to phishing attacks and malware infections, thus enhancing their overall online security.

b) ***User Empowerment*:** By enabling users to independently assess the authenticity of URLs, CheckPhish empowers them to make informed decisions and take proactive measures to protect themselves from online threats.

c) ***Convenience***: CheckPhish offers a user-friendly interface that makes it easy for individuals to quickly verify URLs on various platforms, including emails, websites, and social media, enhancing convenience and accessibility.

Demerits of CheckPhish Website

a) ***False Positives***: CheckPhish may occasionally generate false positives, incorrectly flagging legitimate URLs as phishing attempts, which can lead to user frustration and mistrust in the platform.

b) ***Limited Effectiveness***: While CheckPhish can help identify known phishing URLs, it may not be effective against sophisticated or previously unseen phishing attacks that evade detection by traditional methods.

c) ***Privacy Concerns***: Users may have concerns about privacy when using CheckPhish, particularly if the platform collects and stores data about their browsing habits or interactions with suspicious URLs.

Challenges of CheckPhish Website:

a) ***Adversarial Attacks***: CheckPhish may be vulnerable to adversarial attacks, where malicious actors intentionally manipulate URLs to evade detection by the platform's algorithms, posing a significant challenge to its effectiveness.

b) ***Algorithmic Complexity***: Developing and maintaining effective detection algorithms for CheckPhish requires continuous research and development efforts to keep pace with evolving phishing tactics and techniques.

c) ***User Education***: Ensuring that users understand how to use CheckPhish effectively and interpret its results accurately is crucial for maximizing the platform's impact and reducing the risk of user error or misunderstanding.

d) ***Legal and Ethical Considerations***: CheckPhish must navigate legal and ethical considerations related to data privacy, user consent, and compliance with regulations governing the collection and processing of sensitive information.

e) ***Cross-Platform Compatibility***: Ensuring that CheckPhish is compatible with a wide range of platforms and devices, including desktops, laptops, smartphones, and tablets, presents a technical challenge that must be addressed to maximize its accessibility and utility for users.

2.4 Mail Tester

2.4.1. Introduction

Mail Tester is an email verification tool that can check emails from any database with up to 99% accuracy which is shown in Fig.7 . It can identify duplicate and bounced emails, remove spam traps and emails with high-risk keywords, and verify email domains and syntax.

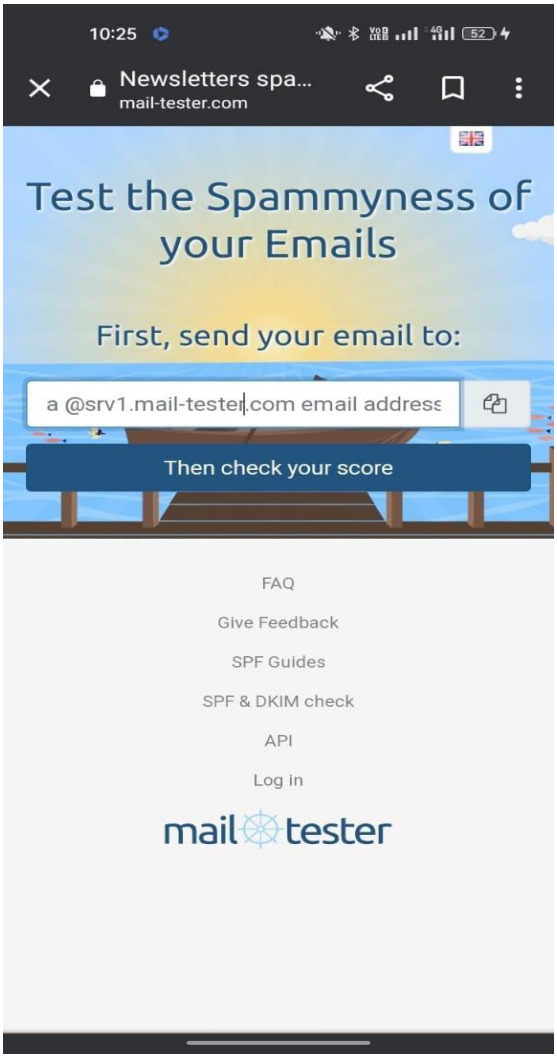

Fig.7: Mail Tester Website

2.4.2. Merits, Demerits and Challenges

Merits of Mail Tester Website

a) ***Email Deliverability Testing***: Mail Tester websites allow users to test the deliverability of their emails by simulating real-world scenarios. This helps users ensure that their emails reach the intended recipients' inboxes without being marked as spam.

b) ***Spam Score Evaluation***: Mail Tester websites provide a spam score for each email tested, indicating the likelihood of the email being flagged as spam by email service providers. This allows users to identify and address potential issues that may trigger spam filters.

c) ***Cost-Effective Solution***: Many Mail Tester websites offer free or affordable testing services, making them accessible to individuals and small businesses with limited budgets.

Demerits of Mail Tester Website

a) ***Limited Accuracy***: While Mail Tester websites provide valuable insights into email deliverability, their tests may not always accurately reflect how email service providers will treat emails in real-world scenarios.

b) ***Dependency on External Factors***: The effectiveness of Mail Tester websites may depend on external factors such as the reputation of the email service provider and the recipient's email settings, which are beyond the control of the user.

c) ***Privacy Concerns*****:** Users may have concerns about privacy and data security when submitting email content to third-party Mail Tester websites, particularly if the content contains sensitive or confidential information.

Challenges of Mail Tester Website

a) ***Adapting to Changing Email Landscape*:** Mail Tester websites must continually adapt their testing methodologies and algorithms to keep pace with evolving email standards, spam detection techniques, and user behaviors.

b) ***Ensuring Accuracy and Reliability*:** Maintaining high levels of accuracy and reliability in test results is crucial for Mail Tester websites to build trust with users and provide actionable insights.

c) ***Handling Diverse Email Content***: Mail Tester websites must be able to accurately analyze and evaluate a wide range of email content types, formats, and languages to provide meaningful feedback to users.

d) ***Balancing User Experience with Complexity*:** Striking the right balance between providing comprehensive testing capabilities and maintaining a user-friendly experience is a challenge for Mail Tester websites, particularly as email testing involves technical concepts that may be unfamiliar to some users.

CHAPTER 3
PROPOSED SOLUTION

CHAPTER 3
PROPOSED SOLUTION

3.1 Objective of Phishing Detection Website

Our proposed system for phishing attacks is a website with comprehensive and multilayered solution designed to enhance online security and protect users from phishing and malware threats. This comprehensive approach involves several layers of analysis to assess the legitimacy of URLs.

The multi-layered approach includes URL analysis, Port forwarding detection, Database comparison, Greek alphabet analysis and Homographic attack detection. The URL which we want to check has to be pasted in the website and the following analysis is done and result is displayed along with the accuracy which is shown in Fig.8 .

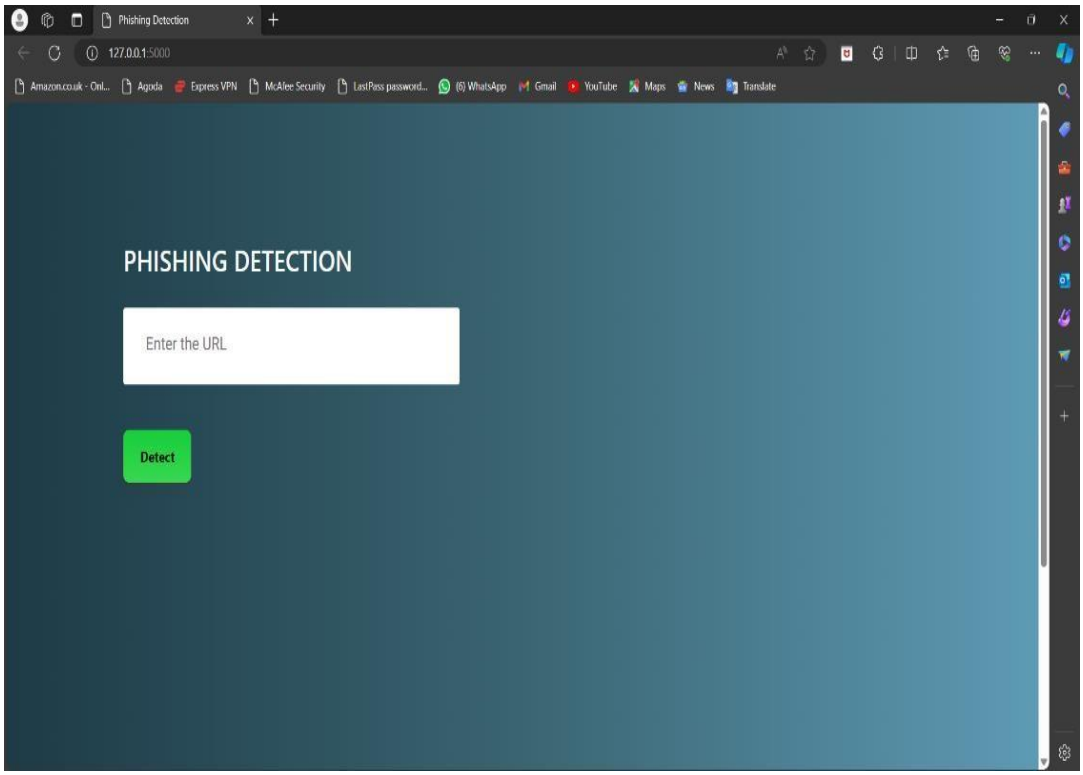

Fig.8: Phishing Detection Website

3.2. Algorithm used for Phishing Detection Website

Step: 1 ***Data Collection***:

- Gather a diverse dataset of URLs labeled as either legitimate or phishing. Sources may include public repositories, APIs, or manual data collection efforts.
- Ensure the dataset is balanced and representative of real-world scenarios.

Step: 2 ***Data Preprocessing***:

- Perform preprocessing steps to clean and prepare the dataset:
- Remove duplicates, irrelevant features, and incomplete entries.
- Extract relevant features from the URLs, such as domain length, presence of HTTPS, and use of suspicious keywords.
- Encode categorical features and scale numerical features as necessary.

Step: 3 ***Model Training***:

- Split the preprocessed dataset into training and testing sets.
- Train multiple machine learning models using a multi-layered approach:
- Train a Decision Tree model on the preprocessed data.
- Train a Random Forest model on the preprocessed data.
- Explore other ensemble methods such as Gradient Boosting or AdaBoost.
- Optionally, train additional models such as Logistic Regression or Support Vector Machines for comparison.
- Evaluate each model's performance on the testing set using metrics like accuracy, precision, recall, and F1-score.

Step: 4 *Comparison*:

- Compare the performance of different machine learning models:
- Analyze the accuracy and other evaluation metrics of each model to determine which performs best for phishing detection.
- Consider the trade-offs between model complexity, interpretability, and performance.
- Conduct statistical tests to assess the significance of performance differences between models.
- Select the best-performing model(s) for deployment in the phishing detection website.

Step: 5 *Deployment*:

- Implement the selected model(s) into the phishing detection website:
- Develop a user-friendly interface for users to input URLs and receive phishing detection results.
- Integrate the model(s) with backend systems to perform real-time phishing detection.
- Ensure scalability and reliability of the website infrastructure to handle user traffic and requests.
- Implement security measures to protect user data and prevent unauthorized access to the system.

Step: 6 *Monitoring and Maintenance*:

- Monitor the performance of the deployed model(s) in production:
- Continuously collect new data to retrain the model(s) and adapt to evolving phishing threats.
- Monitor website usage metrics and user feedback to identify areas for improvement.
- Update the model(s) and website as needed to address performance issues or changes in the phishing landscape.

3.3 Phishing Detection Architecture

The Architecture of Phishing Detection is shown in Fig.9 .

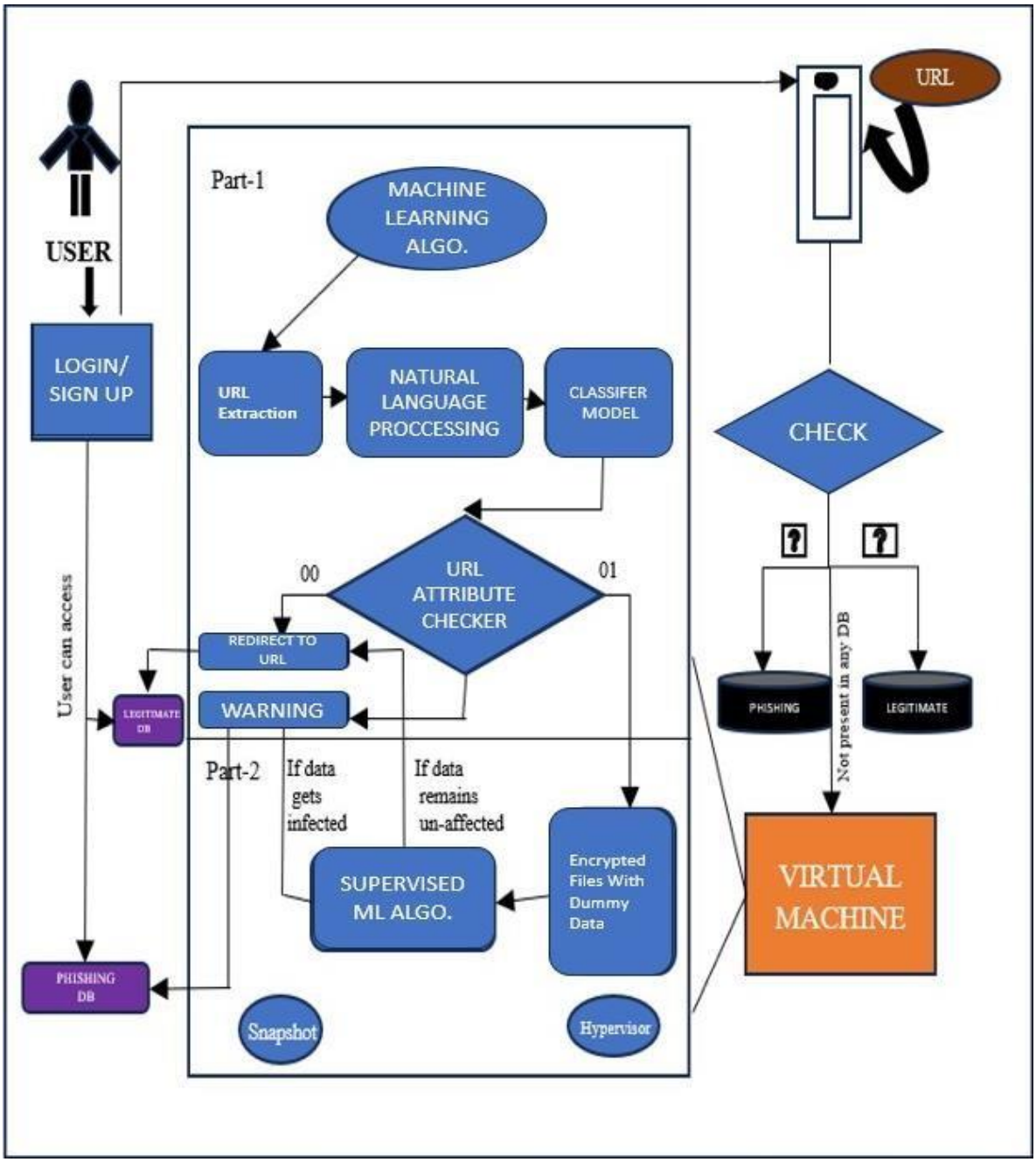

Fig.9: Phishing Detection Architecture

3.4 Stepwise Implementation and Code

a. Loading the data

- The dataset is borrowed from Kaggle, https://www.kaggle.com/eswarchandt/phishing-website-detector .
- A collection of website URLs for 11000+ websites. Each sample has 30 website parameters and a class label identifying it as a phishing website or not (1 or -1).
- The overview of this dataset is, it has 11054 samples with 32 features.

b. Familiarizing with Data & EDA

- In this step, few data frame methods are used to look into the data and its features.

i. Using IP address

Verifying if the URL contains an IP address. Phishing websites frequently use direct IP addresses to conceal their genuine identities.

ii. Long URL

Recognizing unusually lengthy URLs, which are frequently used in phishing efforts to trick people.

iii. Short URL

Examining abbreviated URLs, which are frequently used to hide their destination and can direct users to phishing websites.

iv. Symbol@

Searching for the "@" symbol, as this could signal an attempt to trick consumers by imitating official email correspondence.

v. *Redirecting//*

Recognizing the usage of excessive redirects. It is an tactic that phishing websites frequently utilize to trick visitors.

vi. *Prefix/ Suffix*

Look closely at URL prefixes and suffixes, as phishers may use these to craft phony site addresses.

vii. *Subdomains*

Counting the subdomains is important since an unusually high number could indicate a phishing effort.

viii. *HTTPS*

Check if HTTPS is being used, as many phishing sites do not use secure connections, although genuine websites frequently do.

ix. *Domain Registration Length*

Examining how long a domain has been registered, since shorter periods may be a sign of a fraudulent website.

x. *Favicon*

Verifying the existence and authenticity of the favicon connected to the website, since phishers can trick users with phony favicons.

xi. *Non- Standard Ports*

Identifying non-standard ports used in URLs is important since it might be a sign of phishing activity. This is referred to as a Non-Standard Port.

xii. HTTPS Domain URL

Verifying that, while using HTTPS, the domain and the URL are consistent with each another, as differences could indicate phishing efforts.

xiii. Requested URL

Examining the requested URL's validity because phishing websites may tamper with it to deceive users.

xiv. Anchor URL

Analyzing the anchor links' destination because they may lead to fraudulent websites.

xv. Links in Script Tags

Finding links contained within script tags. It is an tactic that attackers frequently use to conceal harmful URLs.

xvi. Server Form Handler

Examine the form handler on the server for any anomalies in form processing or other possible indications of phishing.

xvii. Information of Email

Recognizing shady email addresses which are connected to the website, since phishing websites could utilize phony contact details.

xviii. Abnormal URL

Recognizing URLs that don't follow standard patterns and indicating possible phishing attempts.

xix. Website forwarding

Recognizing whether a website is sending users to another, since this may be a phishing technique.

xx. Status Bar Cust

Examining changes made to the status bar that could become accustomed to trick users about where a link leads.

xxi. Disabled Right Click

Tracking efforts to turn off the capacity for right-click on web pages. It is the common tactic to keep users from using browser functions that would reveal phishing attempts.

xxii. Using Popup Window

Analyzing the application of popup windows, which phishing websites may employ to divert attention or deceive visitors.

xxiii. Iframe Redirection

Determining whether iframe redirection—a method of sending viewers to phishing pages—is present.

xxiv. Age of Domain

Examining the domain's age is important because younger domains might be more connected to phishing.

xxv. DNS Recording

Verifying any irregularities in DNS recording, since phishers can alter DNS records to trick users.

xxvi. Website traffic analysis

The traffic features of phishing websites frequently diverge from those of trustworthy websites.

xxvii. PageRank

Determining a website's page rank; phishing sites may have artificially inflated or decreased ratings.

xxviii. Google Index

Verifying whether Google has indexed the website since reputable websites have a higher chance of being listed.

xxix. Links Pointing To Page

Analyzing how many external links lead to a particular website to determine its legitimacy or potential for phishing.

xxx. Stats Report

Analyzing the website's statistical reports and searching for any anomalies that might point to phishing activities.

xxxi. class

The class label that, after the previously specified components are looked at, indicates whether or not the situation is considered to be a phishing attempt.

c. Visualizing the data

- Few plots and graphs are displayed to find how the data is distributed and the how features are related to each other which is shown in Fig.10 .

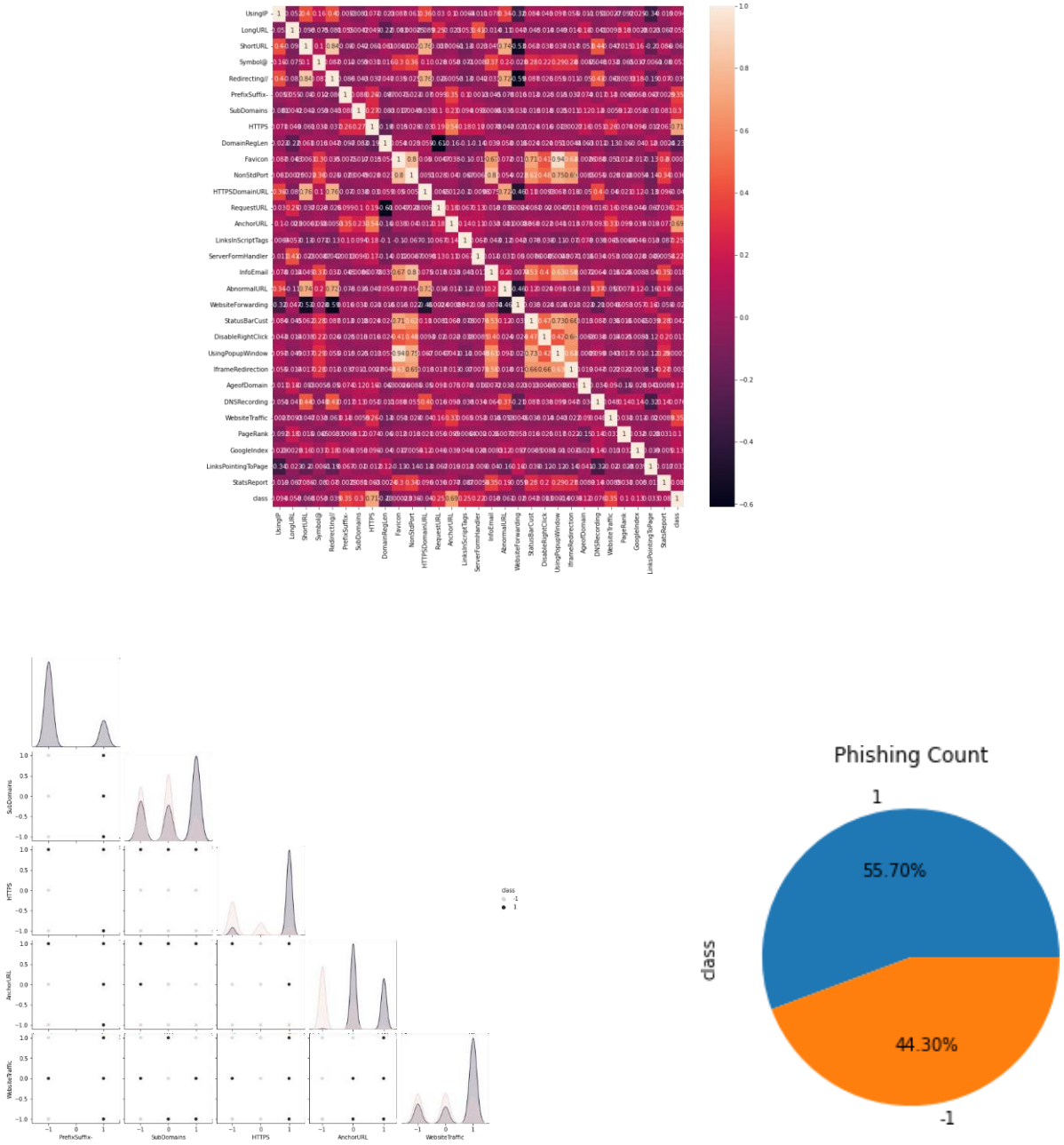

Fig.10 : Data Visualization

d. Splitting the Data

The data is split into train & test sets, 80-20 split.

```
# Splitting the dataset into dependant and independant fetature

X = data. drop(["class"],axis =1)
y = data["class"]

# Splitting the dataset into train and test sets: 80-20 split

from sklearn. model_ selection import train_ test_ split

X_ train, X_ test, y_ train, y_ test = train_ test_ split(X, y, test _size = 0.2, random _state = 42)
X_ train. shape, y_ train. shape, X_ test. shape, y_ test .shape

<<((8843, 30), (8843,), (2211, 30), (2211,))
```

e. Model Building & Training

Supervised machine learning is one of the most commonly used and successful types of machine learning. Supervised learning is used whenever we want to predict a certain outcome/label from a given set of features, and we have examples of features-label pairs. We build a machine learning model from these features-label pairs, which comprise our training set. Our goal is to make accurate predictions for new, never-before-seen data.

There are two major types of supervised machine learning problems, called classification and regression. Our data set comes under regression problem, as the prediction of suicide rate is a continuous number, or a floating-point number in programming terms. The supervised machine learning models (regression) considered to train the dataset in this notebook are:

1. *Logistic Regression*

   Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems.

2. *k-Nearest Neighbors*

   K-Nearest Neighbor is one of the simplest Machine Learning algorithms based on Supervised Learning technique. K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.

3. *Support Vector Classifier*

   Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future.

4. *Naive Bayes*

   Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems.It is mainly used in text, image classification that includes a high-dimensional training dataset. Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

5. *Decision Tree*

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.

6. *Random Forest*

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

7. *Gradient Boosting*

Gradient boosting classifiers are a group of machine learning algorithms that combine many weak learning models together to create a strong predictive model. Decision trees are usually used when doing gradient boosting. Boosting algorithms play a crucial role in dealing with bias variance trade-off. Unlike bagging algorithms, which only controls for high variance in a model, boosting controls both the aspects (bias & variance), and is considered to be more effective.

8. *Catboost*

CatBoost is a recently open-sourced machine learning algorithm from Yandex. It can easily integrate with deep learning frameworks like Google's TensorFlow and Apple's Core ML. It can work with diverse data types to help solve a wide range of problems that businesses face today.

9. *Xgboost*

XGBoost is an implementation of gradient boosted decision trees designed for speed and performance that is dominative competitive machine learning. In this post you will discover how you can install and create your first XGBoost model in Python.

10. *Multilayer Perceptrons*

MLPClassifier stands for Multi-layer Perceptron classifier which in the name itself connects to a Neural Network. Unlike other classification algorithms such as Support Vectors or Naive Bayes Classifier, MLPClassifier relies on an underlying Neural Network to perform the task of classification.

The metrics considered to evaluate the model performance are Accuracy & F1 score.

f. Comparison of Models

To compare the models performance, a data frame is created. The columns of this data frame are the lists created to store the results of the model.

- ***app.py***

```
#importing required libraries

from flask import Flask, request, render_template

import numpy as np

import pandas as pd

from sklearn import metrics

import warnings

import pickle

warnings.filterwarnings('ignore')

from feature import FeatureExtraction

file = open("pickle/model.pkl","rb")

gbc = pickle.load(file)

file.close()

app = Flask(__name__)

@app.route("/", methods=["GET", "POST"])

def index():

    if request.method == "POST":

        url =  request.form["url"]

        obj = FeatureExtraction(url)

        x = np.array(obj.getFeaturesList()).reshape(1,30)

        y_pred =gbc.predict(x)[0]

        #1 is safe
```

```
#-1 is unsafe

y_pro_phishing = gbc.predict_proba(x)[0,0]

y_pro_non_phishing = gbc.predict_proba(x)[0,1]

# if(y_pred ==1 ):

pred = "It is {0:.2f} % safe to go ".format(y_pro_phishing*100)

return render_template('index.html',xx =round(y_pro_non_phishing,2),url=url )

return render_template("index.html", xx =-1)

if__name__== "_main_":

app.run(debug=True)
```

CHAPTER 4
SYSTEM REQUIREMENTS

CHAPTER 4
SYSTEM REQUIREMENTS

4.1 Software Requirements

- ***Operating System*:** Windows 10, 11

Fig.11: Windows

- ***Coding Language* :** Python

Fig.12: Python

- ***Technologies*** **:**

Fig.13 : Scikit Learn, Flask, Pandas**,** NumPy

- ***Editor*** **:** VS Code, Python 3.1

Fig.14: VS Code

Fig.15: Python 3.1

4.2 Hardware Requirements

System **:** Any Processor of 3rd,4th or 5th Generation

Hard Disk **:** Min 50Gb of vacant space

RAM **:** MIN 4GB

CHAPTER 5
RESULTS

CHAPTER 5

RESULTS

5.1. Results

The output of the phishing detection project is shown below in the Fig.16 and Fig.17 .

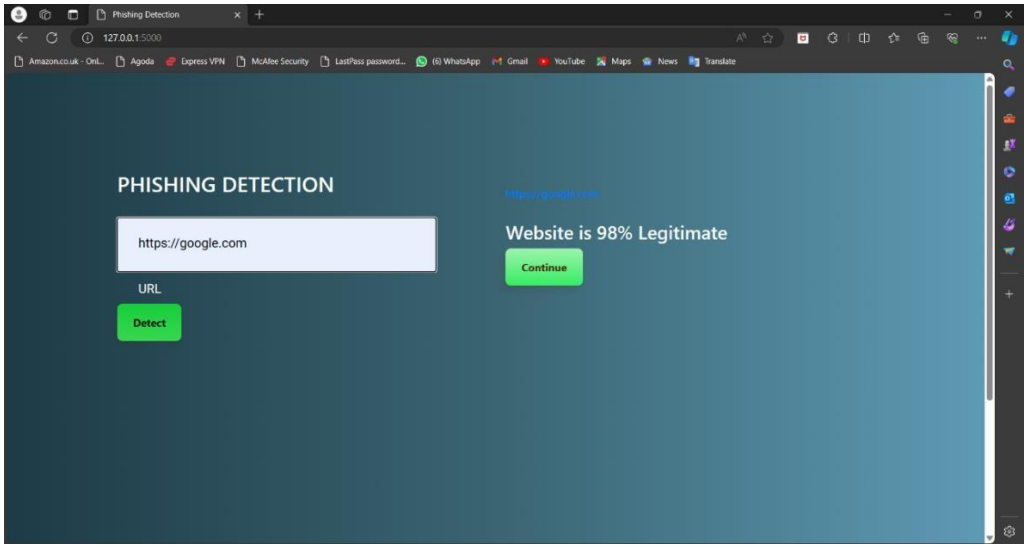

Fig.16: Result of Legitimate URL

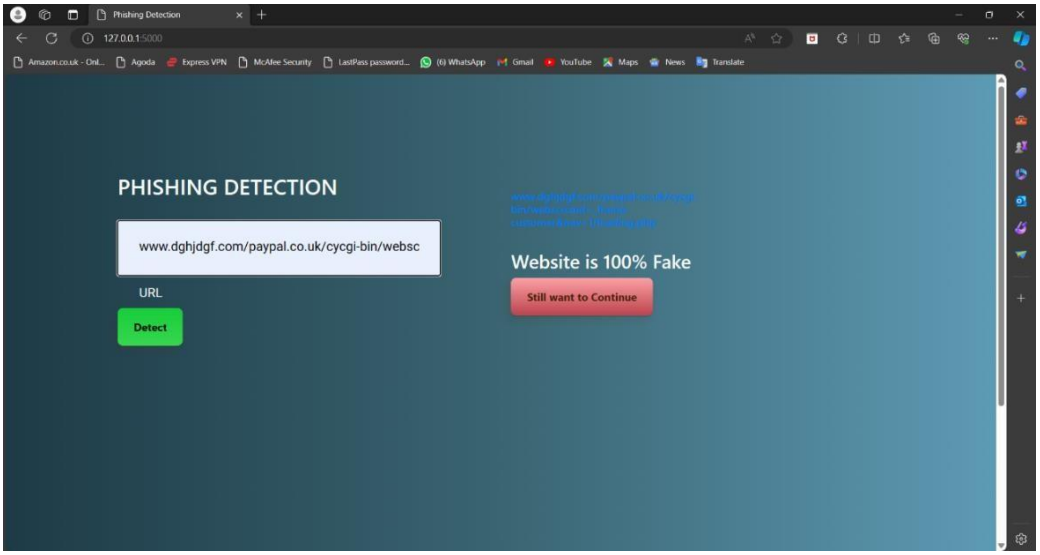

Fig.17: Result of Phishing URL

CHAPTER 6
CONCLUSION

CHAPTER 6
CONCLUSION

6.1 Conclusion and Future Scope

Our project is a major advancement in online security, countering phishing and malware threats with a multi-layered approach. We employ grammar analysis, database comparison, port forwarding detection, and Greek alphabet analysis to thoroughly assess URLs. Our system detects semantic anomalies and evolving cybercriminal techniques, promptly alerting users. We aim to significantly reduce phishing and malware success rates, creating a safer digital environment. Our unwavering commitment to online security empowers users to navigate the internet with confidence. Detecting and mitigating phishing attacks is a critical aspect of ensuring online security and protecting sensitive information. In conclusion, the evolving nature of phishing threats necessitates a multi-faceted approach that combines advanced technological solutions, user education, and ongoing vigilance. While technological tools such as email filtering, website analysis, and machine learning algorithms play a crucial role in identifying and preventing phishing attempts, they are most effective when complemented by user awareness and education programs. Continuous improvement of phishing detection technologies is essential, as threat actors constantly adapt and develop new methods to bypass existing defenses.

Phishing attacks continue to raise serious challenges to cybersecurity, necessitating continuous improvements in detection technologies and the incorporation of new functionality. In this work, a future scope for improving the security of Phishing Detection software, with an emphasis on the development and implementation of unique features such as an Extension for E-mail and Message Applications, User Education initiatives, and an Authentication Page. These enhancements aim to strengthen existing detection techniques while also addressing new attack pathways and user vulnerabilities. Our suggested paradigm stresses the meeting of technology breakthroughs and user-centric techniques to strengthen businesses' resilience to phishing threats.

REFERENCES

REFERENCES

[1] Anti-phishing Working Group (APWG) Phishing Activity Trends Report, 4th quarter 2020, https://docs.apwg.org/reports/apwg trends report q4 2020.pdf

[2] FBI Internet Crime Report 2020, https://www.ic3.gov/Media/PDF/AnnualReport/2020 IC3Report.pdf

[3] Verizon 2020 Data Breach Investigation Report, https://enterprise.verizon.com/resources/reports/2020-databreachinvestigations-report.pdf

[4] World Health Organization, Communicating for Health, Cyber Security, https://www.who.int/about/communications/cyber-security

[5] N. Abdelhamid, A. Ayesh, and F. Thabtah, “Phishing detection-based associative classification data mining,” Expert Systems with Applications, vol. 41, no. 13, pp. 5948–5959, 2014.

[6] K. L. Chiew, E. H. Chang, W. K. Tiong et al., “Utilization of website logos for phishing detection,” Computers & Security, vol. 54, pp. 16–26, 2015.

[7] K. M. Kumar and K. Alekhya, "Identifying fraudulent websites through fuzzy logic," International Journal of Advanced Research in Computer Engineering Technology, 2016.

[8] Rishikesh Mahajan and Irfan Siddavatam, “Phishng website detection using machine learning algorithms,” Internationl Journal of Computer Applications, vol. 181, no. 23, 2018.

[9] Jitendra Kumar, A. Santhanavijayan, B. Janet, Balaji Rajendran, and Bindhumadhava BS, “Phishing website classification and detection using machine learning,” Internationl Conference on Computer Communication and Informatics, 2020

[10] Mehmet Korkmaz, Ozgur Koray Sahingoz, and Banu Diri, "Detection of phishing websites by using machine learning-based URL analysis," 11th Internationl Conference on Computing, Communication, and Networking Technologies , 2020

[11] Mohammad Nazmul Alam, Dhiman Sarma et al., "Phishing attacks detection using a ML approach," 3rd Internationl Conference on Smart Systems and Inventive Technology , 2020

[12] Abdulhamit Subasi, Esraa Molah, Fatin Almkallawi, and Touseef J. Chaudhery, "Intelligent phishing website detection using a Random Forest classifier," International Conference on Electrical and Computing Technologies and Applications (ICECTA), 2017.

[13] https://towardsdatascience.com/phishingdomain-detection-with-ml5be9c99293e5

[14] Rami M. Mohammad, Fadi and Lee McCluskey, "Phishing Website Features,"

RESEARCH PAPER AND CERTIFICATES

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue IV Apr 2024- Available at www.ijraset.com

Phishing Detection: A Multilayer Approach to Scale Down Phishing

Garikapati.Charan Sai[1], Akula.Rishika[2], Garlapati.Dheeraj Reddy[3], Reddyvari Venkateswara Reddy[4], Punyaban Patel[5]
[1, 2, 3]B.Tech Student, [4]Associate Professor, [5] Professor, Department of CSE (Cyber Security), CMR College Of Engineering & Technology, Hyderabad, Telangana, India

Abstract: ***Detection of phishing scams is a process to detect phishing attempts before they happen, inform administrators and users about them, and most importantly reduce the risk. Phishing is a form of cybercrime in which hackers attempt to trick victims into revealing confidential information, e.g. passwords or financial details, by pretending that they're an efficient source for the attack via email and most commonly text messaging. Since phishing is mostly a semantics-based attack that targets human vulnerabilities, identifying these phishing websites can be difficult. The authors suggested this project as a solution to this problem. By identifying and quickly warning users of potential phishing and malware threats, our project aims to improve online security. This all-inclusive method includes Greek alphabet analysis, port forwarding detection, database comparison, grammar analysis utilizing NLP libraries, and many more. A modified version of phishing detection technique has been suggested which integrates Greek alphabet analysis , port forwarding detection and homograph attack detection.***
Keywords: ***text cryptography, text steganography, password strength checker, random strong password generator, information security.***

I. INTRODUCTION

The prevalence of online communication and transactions in the quickly developing information technology age made phishing attacks a growing concern to both individuals and organizations. Cybercriminals employ phishing, a deceitful technique, to manipulate users into disclosing sensitive information. Phishing puts the confidentiality and integrity of personal and organizational data at serious risk. The rising sophistication of approaches utilized by malevolent actors means that traditional security solutions frequently fall short of protecting against these dynamic threats. Therefore, the necessity for sophisticated and adaptable systems that can quickly recognize and stop phishing attacks is imperative. This work presents a complete method of phishing detection that combines behavioral analysis, domain-specific heuristics, and state-of-the-art machine learning techniques. By combining these components, our suggested system seeks to improve the precision and effectiveness of spotting phishing attempts on different websites, social media platforms, and emails.

Our suggested solution is intended to improve users' internet security by identifying and swiftly informing them of potential harmful programs and threats. This all-inclusive method includes Greek alphabet analysis, database comparison, port forwarding detection using Python tools, grammar analysis using NLP libraries, and many more.

II. LITERATURE REVIEW

Three techniques were proposed for categorization from the writers of the paper [8], Rishikesh Mahajan and Irfan Siddavatam: Decision Tree, Random Forest, and Support Vector Machine. Their sample included 19,653 phishing URLs and 17,058 benign URLs, each with 16 attributes, collected from the websites of Alexa and PhishTank, respectively. There were two sections in the data sets: identifying and training in the following ratios: 50:50, 70:30, and 90:10, respectively. The rates of false positive and negative and accuracy score were among the metrics used to assess performance. They achieved 97.14% accuracy using the Random Forest algorithm as it has low false negative rate. The study found that increasing the quantity if the training data gest better accuracy. Jitendra Kumar et al.'s study from [9] trained a variety of classifiers, including K Nearest Neighbour, Random Foresst, Naive Bayees Classifier, Logistic Regression, and Decision Trees, using variables extracted from the lexical structure of the URL. The URL dataset was created with volatility, overfitting, biased training, and unbalanced data in mind. The dataset had an equal number of tagged phishing and genuine URLs and was further split into testing and training groups using a 7:3 ratio. Even though the AUC values among all the categorization were almost equal, the Naive Bayees Classifier turned out to be more suitable because it had the highest value.

The most accuracy that Naive Bayes could attain was 98%, with precision=1, recall=0.95, and F1-score= 0.97. Mehmet Korkmaz et al. developed a detection of phishing scams by ML in [10] by utilising eight different algorithms on three different datasets.
The techniques used were XGBoost, Random Forest (RF), Naive Bayes (NB), K-Nearest Neighbour (KNN), Suport Vector Machine (SVM), Logistic Regression (LR), K-Nearest Neighbour (KNN), and Artificial Neural Network (ANN). It was discovered that the models using LR, SVM, and NB had low accuracy rates.
The techniques used were XGBoost, Random Forest (RF), Naive Bayes (NB), K-Nearest Neighbour (KNN), Suport Vector Machine (SVM), Logistic Regression (LR), K-Nearest Neighbour (KNN), and Artificial Neural Network (ANN). It was discovered that the models using LR, SVM, and NB had low accuracy rates.
The 32-feature Kaggle dataset underwent principal component analysis (PCA) and various feature selection procedures. The dataset benefits from feature selection since it eliminates unnecessary or redundant data. The proposed model used the REF, Relief-F, IG, and GR algorithms for feature selection before turning to PCA. Random Forest achieved an accuracy rate of 97%.
It handled the over-fitting issue better and was less erratic. Using the UCI dataset, Abdulhamit Subasi et al. launched an intelligent phishing detection software in [12]. To recognize phishing websites, a numerous machine learning strategies were used among the classifiers are Artificial Neural Networks (ANN), K-Nearest Neighbours (K-NN), Support Vector Machines (SVM), C4.5 Decision Trees, Random Forests (RF), and Rotation Forests (RoF).

III. METHODOLOGY

A website that enables visitors to thoroughly examine the legitimacy of the URL using multiple layers of analysis. Greek Alphabet Analysis, Database Comparison, Grammar Analysis, and Port Forwarding Detection are among the procedures.

1) Analysis of Grammar

Grammar analysis looks for irregularities or inconsistencies in the linguistic structure of textual information. Grammar analysis detects dishonest language patterns suggestive of fraudulent conduct, which improves the accuracy of phishing attempt detection.

- *Input:* A designated text box is filled with the URL to be evaluated.
- *Process:* Using preset rules and patterns, the algorithm thoroughly analyses the URL's grammar. This investigation looks for any irregularities or deviations that might point to a phishing effort or malevolent intent.
- *Output:* A result indicating whether or not the URL adheres to predicted grammatical patterns is produced by the grammar analysis.

2) Database Comparison

Database comparison is the technique of comparing features and patterns between suspicious data and pre-existing phishing databases to find fraudulent information more rapidly. As a result, detection of phishing software are more successful overall.

- *Input*: The URL under scrutiny is compared against a comprehensive database of known legitimate links and a database of verified phishing links.
- *Process*: The system utilizes predefined databases to check if the entered URL matches any known legitimate or phishing URLs. This database comparison helps identify links with a history of being associated with phishing attacks or legitimate entities.
- *Output*: The result provides information on whether the URL matches any entries in the legitimate or phishing databases.

3) Port Forwarding Detection

Monitoring network traffic for instances of port forwarding is referred to as port forwarding detection, and it is an essential component of phishing detection since it can reveal any attempts to reroute users to malicious servers, strengthening defenses against phishing assaults.

- *Input*: A phishing assault exploitable technique is detected by analyzing the URL.
- *Process*: The system looks for port forwarding, which is the procedure for diverting network traffic from one port to another. This method can be used by attackers to hide dangerous URLs.
- *Output*: The result alerts the user to a possible security risk by indicating whether port forwarding is found in examined URL.

4) Greek Alphabet Analysis

Greek alphabet analysis in phishing detection entails searching for Greek characters in the text since hackers commonly use them to circumvent standard filters. The ability of the system to identify and neutralize phishing threats is enhanced by locating these anomalies.

- *Input*: An examination of the URL is carried out using an emphasis on how the Greek alphabet is utilized in the URL structure
- *Procedure*: The system looks for Greek letters in the URL because hackers could use them to make visually appealing URLs that trick visitors.
- *Output*: The result indicates if the URL contains any Greek alphabet characters, indicating a possible attempt to build a bogus URL.

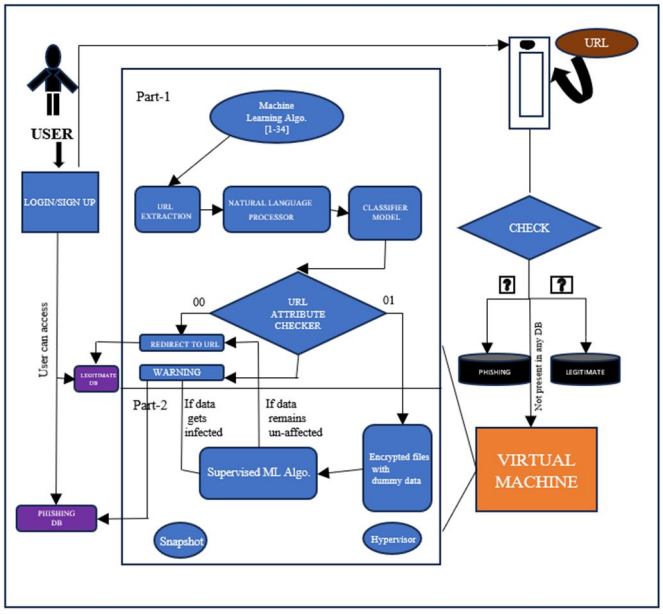

Fig.1- Architecture of Phishing Detection

5) Using IP address

Verifying if the URL contains an IP address. Phishing websites frequently use direct IP addresses to conceal their genuine identities.

6) Long URL

Recognizing unusually lengthy URLs, which are frequently used in phishing efforts to trick people.

7) Short URL

Examining abbreviated URLs, which are frequently used to hide their destination and can direct users to phishing websites.

8) Symbol@

Searching for the "@" symbol, as this could signal an attempt to trick consumers by imitating official email correspondence.

9) Redirecting//

Recognizing the usage of excessive redirects. It is an tactic that phishing websites frequently utilize to trick visitors.

10) Prefix/ Suffix

Look closely at URL prefixes and suffixes, as phishers may use these to craft phony site addresses.

11) Subdomains

Counting the subdomains is important since an unusually high number could indicate a phishing effort.

12) HTTPS
Check if HTTPS is being used, as many phishing sites do not use secure connections, although genuine websites frequently do.

13) Domain Registration Length
Examining how long a domain has been registered, since shorter periods may be a sign of a fraudulent website.

14) Favicon
Verifying the existence and authenticity of the favicon connected to the website, since phishers can trick users with phony favicons.

15) Non- Standard Ports
Identifying non-standard ports used in URLs is important since it might be a sign of phishing activity. This is referred to as a Non-Standard Port.

16) HTTPS Domain URL
Verifying that, while using HTTPS, the domain and the URL are consistent with each another, as differences could indicate phishing efforts.

17) Requested URL
Examining the requested URL's validity because phishing websites may tamper with it to deceive users.

18) Anchor URL
Analyzing the anchor links' destination because they may lead to fraudulent websites.

19) Links in Script Tags
Finding links contained within script tags. It is an tactic that attackers frequently use to conceal harmful URLs.

20) Server Form Handler
Examine the form handler on the server for any anomalies in form processing or other possible indications of phishing.

21) Information of Email
Recognizing shady email addresses which are connected to the website, since phishing websites could utilize phony contact details.

22) Abnormal URL
Recognizing URLs that don't follow standard patterns and indicating possible phishing attempts.

23) Website forwarding
Recognizing whether a website is sending users to another, since this may be a phishing technique.

24) Status Bar Cust
Examining changes made to the status bar that could become accustomed to trick users about where a link leads.

25) Disabled Right Click
Tracking efforts to turn off the capacity for right-click on web pages. It is the common tactic to keep users from using browser functions that would reveal phishing attempts.

26) Using Popup Window
Analyzing the application of popup windows, which phishing websites may employ to divert attention or deceive visitors.

27) Iframe Redirection
Determining whether iframe redirection—a method of sending viewers to phishing pages—is present.

28) Age of Domain
Examining the domain's age is important because younger domains might be more connected to phishing.

29) DNS Recording
Verifying any irregularities in DNS recording, since phishers can alter DNS records to trick users.

30) Website traffic analysis
The traffic features of phishing websites frequently diverge from those of trustworthy websites.

31) PageRank
Determining a website's page rank; phishing sites may have artificially inflated or decreased ratings.

32) Google Index
Verifying whether Google has indexed the website since reputable websites have a higher chance of being listed.

33) Links Pointing To Page
Analyzing how many external links lead to a particular website to determine its legitimacy or potential for phishing.

34) Stats Report
Analyzing the website's statistical reports and searching for any anomalies that might point to phishing activities.

35) class
The class label that, after the previously specified components are looked at, indicates whether or not the situation is considered to be a phishing attempt.

IV. USER INTERFACE

User Interaction: The user hits the submit button to obtain the outcome after the analytic procedures. - Output: The user sees a full result that summarizes the information gleaned from every level of investigation. This contains details on the URL's grammatical accuracy, database presence, port forwarding detection, and Greek character usage. To raise the identification of phishing attempts, this methodology offers a multidimensional approach to URL legitimacy assessment by integrating linguistic analysis, historical database comparison, and unique threat indicators.

V. RESULT & DISCUSSIONS

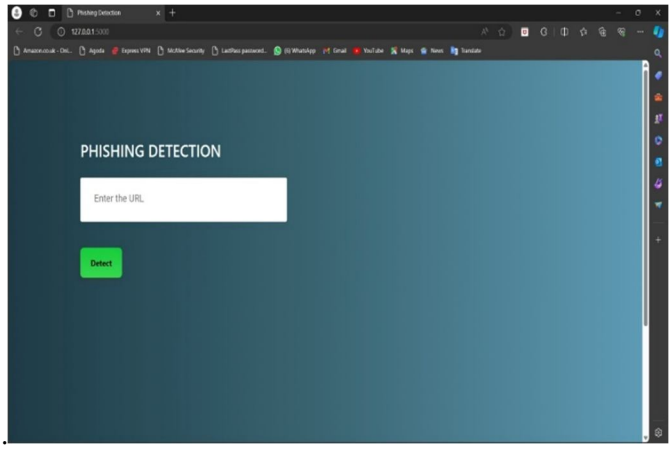

Fig.2 – User Interface

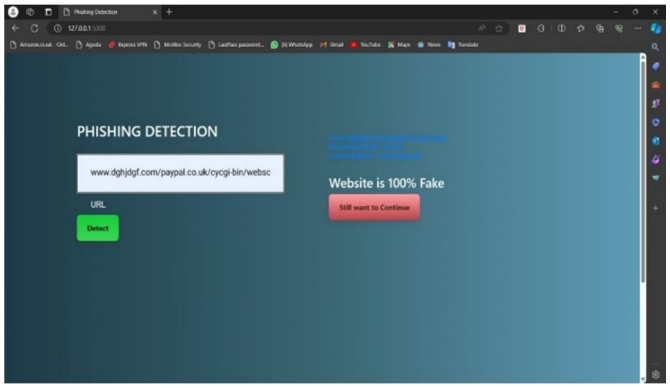

Fig.3- Detection of Phishing URL

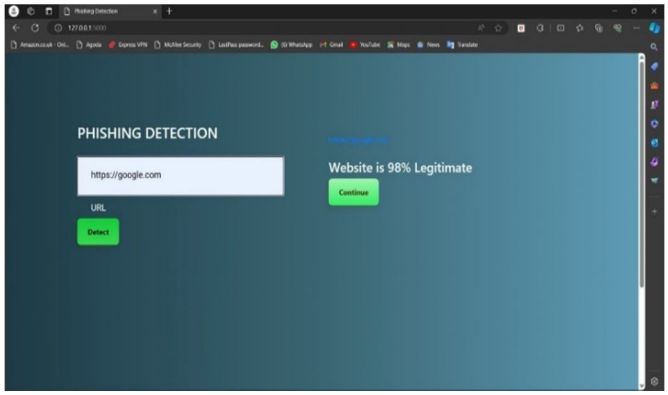

Fig.4- Detection of Legitimate URL

	ML Model	Accuracy	f1_score	Recall	Precision
0	Gradient Boosting Classifier	0.974	0.977	0.994	0.986
1	CatBoost Classifier	0.972	0.975	0.994	0.989
2	XGBoost Classifier	0.969	0.973	0.993	0.984
3	Multi-layer Perceptron	0.969	0.973	0.995	0.981
4	Random Forest	0.967	0.971	0.993	0.990
5	Support Vector Machine	0.964	0.968	0.980	0.965
6	Decision Tree	0.960	0.964	0.991	0.993
7	K-Nearest Neighbors	0.956	0.961	0.991	0.989
8	Logistic Regression	0.934	0.941	0.943	0.927
9	Naive Bayes Classifier	0.605	0.454	0.292	0.997

Fig.5-Accuracy of various models used for URL detection

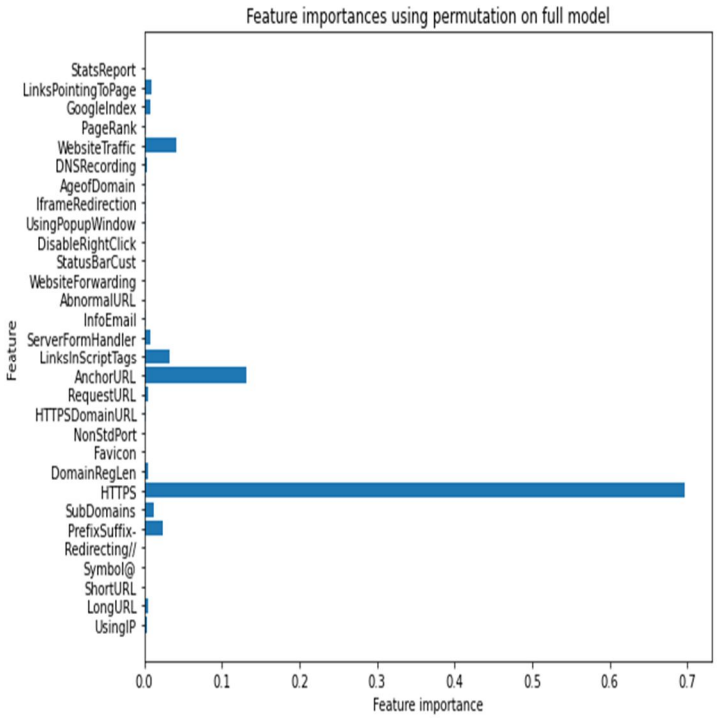

Fig.6- Feature importance for Phishing URL Detection

VI. BENEFITS

These are the benefits for using a multi-layered approach for Phishing detection are

A. *More Accurate Detection*

1) *Smart Technology*: We employ state-of-the-art technologies like machine learning to improve accuracy. This facilitates the system's gradual adaptability as it picks up new tricks for phishing.
2) *Real-time Monitoring*: The system keeps an eye on user actions within real-time, quickly spotting potential phishing threats and providing timely alerts.

B. *Detecting Unknown Attacks*

1) *Adaptive Systems*: The system is designed to detect new and unknown phishing attacks by looking for unusual patterns in user behavior.
2) *Smart Guesswork*: Heuristic methods help the system make educated guesses regarding potential threats, even if they're brand new or haven't been seen before.

C. *Alerting Users*

1) *Clear Warnings*: When a threat is detected, the system sends clear and user-friendly warnings. These alerts help users understand the potential danger.
2) *User Preferences*: Users can personalize their alert settings, ensuring that warnings are delivered in a way that suits their preferences.

D. *Multi-Layered Approach*

1) *Many Checks*: The system doesn't rely on a single method. It combines different ways of looking for phishing signs, including checking content, URLs, headers, and sender details.
2) *Double-Check Strategy*: By using both known patterns and analyzing behavior, the system becomes more reliable. This multi-layered approach minimizes mistakes and makes the system robust against different phishing tactics.

VII. CONCLUSION

In conclusion, our project is a noteworthy leap forward in online security by introducing a multi-layered approach to counter phishing and malware threats. By incorporating advanced techniques to detect semantic anomalies and staying vigilant against evolving cybercriminal strategies, our system ensures a robust defense mechanism. The prompt alerting of users adds an extra layer of protection, empowering individuals to make well briefed decisions in real time. With the overarching goal of substantially reducing success rates for both phishing and malware attacks, our contribution aims to establish a safer digital environment. This creative strategy not only addresses current threats but also positions our system to adapt and evolve alongside the dynamic landscape of cyber threats, making a valuable contribution to the ongoing efforts to enhance online security.

VIII. FUTURE SCOPE

Phishing attacks continue to raise serious challenges to cybersecurity, necessitating continuous improvements in detection technologies and the incorporation of new functionality. In this work, a future scope for improving the security of Phishing Detection software, with an emphasis on the development and implementation of unique features such as an Extension for E-mail and Message Applications, User Education initiatives, and an Authentication Page. These enhancements aim to strengthen existing detection techniques while also addressing new attack pathways and user vulnerabilities. Our suggested paradigm stresses the meeting of technology breakthroughs and user-centric techniques to strengthen businesses' resilience to phishing threats.

REFERENCES

[1] Anti-phishing Working Group (APWG) Phishing Activity Trends Report, 4th quarter 2020, https://docs.apwg.org/reports/apwg trends report q4 2020.pdf
[2] FBI Internet Crime Report 2020, https://www.ic3.gov/Media/PDF/AnnualReport/2020 IC3Report.pdf
[3] Verizon 2020 Data Breach Investigation Report, https://enterprise.verizon.com/resources/reports/2020-databreachinvestigations-report.pdf
[4] World Health Organization, Communicating for Health, Cyber Security, https://www.who.int/about/communications/cyber-security
[5] N. Abdelhamid, A. Ayesh, and F. Thabtah, “Phishing detection-based associative classification data mining,” Expert Systems with Applications, vol. 41, no. 13, pp. 5948–5959, 2014.
[6] K. L. Chiew, E. H. Chang, W. K. Tiong et al., “Utilization of website logos for phishing detection,” Computers & Security, vol. 54, pp. 16–26, 2015.
[7] "Identifying fraudulent websites through fuzzy logic," K. M. Kumar and K. Alekhya, International Journal of Advanced Research in Computer Engineering Technology, 2016.
[8] Rishikesh Mahajan and Irfan Siddavatam, “Phishng website detection using machine learning algorithms,” Internationl Journal of Computer Applications, vol. 181, no. 23, 2018.
[9] Jitendra Kumar, A. Santhanavijayan, B. Janet, Balaji Rajendran, and Bindhumadhava BS, “Phishing website classification and detection using machine learning,” Internationl Conference on Computer Communication and Informatics, 2020
[10] Mehmet Korkmaz, Ozgur Koray Sahingoz, and Banu Diri, “Detection of phishing websites by using machine learning-based URL analysis,” 11th Internationl Conference on Computing, Communication, and Networking Technologies , 2020
[11] Mohammad Nazmul Alam, Dhiman Sarma et al., “Phishing attacks detection using a ML approach,” 3rd Internationl Conference on Smart Systems and Inventive Technology , 2020
[12] Abdulhamit Subasi, Esraa Molah, Fatin Almkallawi, and Touseef J. Chaudhery, “Intelligent phishing website detection using a Random Forest classifier,” International Conference on Electrical and Computing Technologies and Applications (ICECTA), 2017.
[13] Structure of a URL: image, https://towardsdatascience.com/phishingdomain-detection-with-ml5be9c99293e5
[14] Rami M. Mohammad, Fadi and Lee McCluskey, “Phishing Website Features,”

ISSN No. : 2321-9653

iJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

Certificate

It is here by certified that the paper ID : IJRASET59828, entitled

Phishing Detection: A Multilayer Approach to Scale Down Phishing

by

Garikapati Charan Sai

after review is found suitable and has been published in
Volume 12, Issue IV, April 2024
in

International Journal for Research in Applied Science & Engineering Technology
(International Peer Reviewed and Refereed Journal)
Good luck for your future endeavors

Editor in Chief, iJRASET

ISRA Journal Impact Factor: 7.429

45.98 INDEX COPERNICUS

THOMSON REUTERS Researcher ID: N-9681-2016

10.22214/IJRASET doi cross ref

TOGETHER WE REACH THE GOAL SJIF 7.429

ISSN No. : 2321-9653

iJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

ISRA Journal Impact Factor: 7.429

45.98 INDEX COPERNICUS

THOMSON REUTERS Researcher ID: N-9681-2016

10.22214/IJRASET doi cross ref

TOGETHER WE REACH THE GOAL SJIF 7.429

Certificate

It is here by certified that the paper ID : IJRASET59828, entitled

Phishing Detection: A Multilayer Approach to Scale Down Phishing

by

Akula Rishika

after review is found suitable and has been published in Volume 12, Issue IV, April 2024

in

International Journal for Research in Applied Science & Engineering Technology

(International Peer Reviewed and Refereed Journal)

Good luck for your future endeavors

Editor in Chief, iJRASET

ISSN No. : 2321-9653

iJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

ISRA Journal Impact Factor: 7.429

45.98 INDEX COPERNICUS

THOMSON REUTERS Researcher ID: N-9681-2016

10.22214/IJRASET

TOGETHER WE REACH THE GOAL

SJIF 7.429

Certificate

It is here by certified that the paper ID : IJRASET59828, entitled

Phishing Detection: A Multilayer Approach to Scale Down Phishing

by

Garlapati Dheeraj Reddy

after review is found suitable and has been published in
Volume 12, Issue IV, April 2024
in

International Journal for Research in Applied Science & Engineering Technology
(International Peer Reviewed and Refereed Journal)
Good luck for your future endeavors

Editor in Chief, iJRASET

ISSN No. : 2321-9653

iJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

ISRA Journal Impact Factor: 7.429

45.98 INDEX COPERNICUS

THOMSON REUTERS Researcher ID: N-9681-2016

10.22214/IJRASET

TOGETHER WE REACH THE GOAL

SJIF 7.429

Certificate

It is here by certified that the paper ID : IJRASET59828, entitled

Phishing Detection: A Multilayer Approach to Scale Down Phishing

by

Reddyvari Venkateswara Reddy

after review is found suitable and has been published in Volume 12, Issue IV, April 2024

in

International Journal for Research in Applied Science & Engineering Technology

(International Peer Reviewed and Refereed Journal)

Good luck for your future endeavors

Editor in Chief, iJRASET

ISSN No. : 2321-9653

iJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

ISRA Journal Impact Factor: 7.429

45.98 INDEX COPERNICUS

THOMSON REUTERS Researcher ID: N-9681-2016

10.22214/IJRASET doi cross ref

TOGETHER WE REACH THE GOAL SJIF 7.429

Certificate

It is here by certified that the paper ID : IJRASET59828, entitled

Phishing Detection: A Multilayer Approach to Scale Down Phishing

by

Punyaban Patel

after review is found suitable and has been published in Volume 12, Issue IV, April 2024

in

International Journal for Research in Applied Science & Engineering Technology

(International Peer Reviewed and Refereed Journal)

Good luck for your future endeavors

Editor in Chief, iJRASET