Fortfying The Digital Landscape Publishing URL Detection

[1] **Mr.P.Naveen Kumar** (Assistant Professor),

Arr.Rohit [2], H.Shirisha [3], M.Chaithanya [4]

[2]ARR.rohit Department of Computer Science and Engineering (Joginpally B.R Engineering College)
[3]H.Shirisha Department of Computer Science and Engineering (Joginpally B.R Engineering College)
[4] M.Chaithanya Department of Computer Science and Engineering (Joginpally B.R Engineering College)

ABSTRACT

The increasing prevalence of cyber threats and malicious activities in the digital landscape has underscored the need for effective detection and prevention mechanisms, particularly in the context of malicious URLs. This project aims to strengthen digital security by detecting malicious URLs, a common tool for cyber threats such as phishing, malware, and data breaches. With the surge in cyberattacks, traditional blacklist-based and signature detection methods struggle to identify new, sophisticated malicious links in real time. To address this, our project leverages machine learning techniques to classify URLs as safe or harmful, offering a proactive approach to user protection. Our system analyze URLs by extracting key features, including length, special characters, domain age, and lexical patterns. It combines these with host-based attributes to detect malicious tendencies. We enhance detection performance through ensemble learning and continuous model optimization using real-world data. This approach enables our solution to identify potential threats instantly, helping to secure users from interacting with harmful URLs. By offering an effective layer of defense against evolving threats, our project contributes to cybersecurity advancements, reinforcing the digital landscape and improving online safety for individuals and organizations alike. This work contributes to the ongoing development of safer digital environments by offering actionable insights and practical solutions for enhancing URL security protocols

1.INTRODUCTION

With the internet playing a central role in our everyday lives—whether it's for communication, business, education, or health services—our reliance on digital platforms is greater than ever. However, this increasing dependence has also opened the door to growing cyber threats. One of the most dangerous and widespread threats in today's digital world is phishing. Phishing attacks usually involve fake websites and deceptive links that appear to be legitimate. These malicious URLs trick users into giving away sensitive information like passwords, bank details, or personal data. What makes phishing particularly dangerous is how convincing these fake sites can look, often making it difficult to identify them with the naked eye or using basic security tools. Although traditional security methods like URL blacklists and signature-based detection exist, they are often one step behind. They typically rely on databases of known threats, which means they struggle to detect new or cleverly disguised phishing links that haven't been seen before.This project aims to solve that problem by using machine learning to build a smarter, more responsive phishing detection system. The system will examine different aspects of a URL—such as its structure, text patterns, and hosting details—to determine whether it's safe or harmful. By constantly learning from new data, the system will be able to detect threats in real time and help protect users from phishing and other online attacks.

1.1 Problem Statement

The rapid expansion of internet connectivity and digital services has brought numerous advantages to individuals, businesses, and institutions across the globe. However, this surge in online activity has also led to a dramatic increase in cyber threats—most notably, phishing attacks that manipulate malicious URLs to deceive users. These URLs are carefully crafted to mimic legitimate websites, making them highly effective at tricking users into revealing sensitive information such as passwords, credit card numbers, and personal identification details.
Conventional phishing detection methods, including URL blacklists and rule-based filtering systems, have long served as the frontline defense against these attacks. While these approaches provide some level of protection, they come with serious limitations. Primarily, they depend on previously identified malicious URLs or

predefined patterns, making them inherently **reactive** rather than **proactive**. As a result, they struggle to detect **zero-day threats**, **obfuscated URLs**, or **dynamically generated phishing links**—all of which are commonly used by modern cybercriminals to evade detection.

This inability to keep pace with evolving threats creates a significant vulnerability within digital systems. Attackers exploit this gap by continuously adapting their tactics, launching sophisticated phishing campaigns that bypass traditional security measures and reach unsuspecting users. The consequences of a successful phishing attack can be severe, including financial loss, data breaches, identity theft, compromised systems, and long-term reputational damage to both individuals and organizations.

Therefore, the core challenge lies in building a **smart, adaptive, and automated detection system** that can accurately identify new and previously unseen phishing URLs in real time. Such a system must not only detect threats with high precision but also minimize false positives to avoid disrupting legitimate user activity. Addressing this problem is essential to strengthening cybersecurity and reducing the growing impact of phishing in our increasingly connected digital world.

1.2 Purpose

The proposed system introduces a comprehensive and adaptive approach to detecting phishing URLs by integrating traditional blacklist filtering with a machine learning-based classification model. Initially, the system checks incoming URLs against a regularly updated blacklist that contains known malicious links. If a match is found, the URL is immediately flagged and blocked, offering a fast and efficient way to handle already identified threats. However, recognizing that many phishing URLs are newly created or designed to evade blacklists, the system includes a second layer of analysis powered by a Random Forest machine learning model. This model examines a variety of features such as URL length, use of special characters, domain structure, presence of IP addresses, HTTPS usage, domain age, and other lexical or structural patterns. The Random Forest algorithm, known for its high accuracy and robustness, uses these features to classify the URL as either safe or harmful. What sets this system apart is its dynamic nature—the blacklist is frequently updated with threat intelligence from trusted sources, and the machine learning model is continuously trained on new data to keep pace with evolving phishing techniques. By combining the speed of blacklist checks with the adaptability and intelligence of machine learning, the proposed system offers a proactive and reliable defense mechanism that effectively mitigates the risks posed by both known and emerging phishing threats.

1.3 Scope

The scope of URL phishing detection encompasses the development and implementation of advanced algorithms and techniques to identify and mitigate malicious activities conducted through deceptive URLs. It involves real-time detection, leveraging machine learning and AI, behavioural analysis, and integration with web browsers and security tools. Additionally, it includes user education, cross-platform compatibility, reducing false positives, compliance with regulations, and continuous monitoring and updates. Overall, the scope aims to provide comprehensive protection against phishing attacks, ensuring the security of users' personal information and online transactions

2. LITERATURE REVIEW

The purpose of the literature review is to give a summary of facts and findings on this project. This can be done by studying or finding the references or related findings. This review will give a better understanding to the need for this project and also help in designing the methodology for this project. In this project methodology section will describe in detail about the selected methodology or approach that will be used in this project. By selecting the suitable methodology, the productivity and quality of the project will be increased and improved. Besides that, the description of the existing system and review of related works are included in this chapter.

BasePaper-1

Title : Identification and Analysis of Phishing Website based on Machine Learning Methods

Author : Mohammed Hazim Alkawaz, Stephanie Joanne Steven form University of Southampton Malaysia (2021)

Objective: • In this paper, authors proposed a hybrid approach which includes a grouping of content similarity whitelists, style similarity and heuristics which is called Phish-Alert.

• Phish-Alert executed more effectively on experimental dataset that includes 500 phishing sites and 500 valid sites.

Scope: • Here is 2 level of process. In the first level is founded on RDD model of webpages and second founded on machine learning technique. Both levels collaborate to reduce the number of false positives to improvise the accuracy and precision of system.12

Disadvantages: • The performance of the Phish-Alert model decreases with an increase in dataset size.

Result: • Among the machine learning algorithms, Random Forest has shown best performed in classification where has good accuracy even with less sensitive outliers and missing values in parameter choices.

Base paper:2

Title : Detecting Phishing Websites Using Machine Learning

Author : S. Alrefaai, G. Ozdemir, A. Mohamed, Detecting Phishing Websites Using Machine Learning, in Proceedings of the 2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, Turkey (2022)

Objective: The objective of this paper is to investigate and propose machine learning techniques for effectively identifying phishing websites...

Scope: i. Investigating the application of machine learning techniques for the detection of phishing websites

. ii. Exploring different features and algorithms used in machine learning models for phishing detection

. iii. Evaluating the effectiveness and accuracy of machine learning- based approaches in identifying phishing websites.

Disadvantage: The main disadvantage is that it delays the processing of the link, as there is a latency ensure users have received the same link and personalized phishing links will not be scanned.

Result:13 The paper focuses on technical defense strategies against phishing attacks, particularly reviewing recent advancements. It highlights the importance of identifying phishing websites and discusses the emergence of machine learning-based approaches for more accurate predictions. The primary goal is to explore effective real-time methods for preventing phishing attacks.

3. SYSTEM ARCHITECTURE

1. Input Layer

- **URL Submission**: The system receives a URL from the user or external system.

2. Preprocessing Layer

- **Feature Extraction**:
  - Extract lexical features (length, number of dots, presence of special characters)
  - Host-based features (domain age, IP-based, DNS records)
  - Content-based features (HTML, JavaScript analysis)
- **Normalization**:
  - Normalize the feature values for uniformity.

3. Blacklist Checker

- **Initial Filtering**:
  - Check the input URL against the **Blacklist Database**.
  - If matched with known phishing entries → **Block and Alert**.
  - If not matched → **Proceed to Classification**.

4. Classification Engine (Machine Learning Layer)

- **Multiple Classifiers Used** (could be implemented as ensemble or individual):
  - K-Nearest Neighbors (KNN)
  - Kernel SVM
  - Decision Tree
  - Random Forest (Bagging)
  - Gradient Boosting (Boosting)
- **Voting Mechanism (Ensemble)**:
  - Combine predictions using majority vote or weighted vote to make final classification.

5. Output Layer

- **Prediction Result**:
  - Label URL as **Phishing** or **Legitimate**
  - Confidence score or probability may also be provided.
- **Action Taken**:
  - Alert user if phishing.
  - Allow access if legitimate.

6. Feedback & Learning Loop

- **Update Models**:
  - Use misclassified instances for model retraining.
- **Blacklist Maintenance**:
  - Add confirmed phishing URLs.
  - Periodically remove outdated entries.

4. SYSTEM REQUIREMENTS

4.1 Hardware Requirement:

• **Processor CPU** - Intel Pentium Dual Core and Higher
• **Hard Disk capacity** - 512MB Space required minimum
• **RAM** - 4GB minimum

4.2 Software Requirements:

• **Operating system** - Windows8 or Above.
• **Coding Language** - Python
• **Data Base** – MySQL
• **Editor**-Visual Studio Code(Vs code)

5 MODELING AND ANALYSIS

5.1. System Modeling

The system design phase serves as the bridge between requirements gathering and system implementation. It transforms user requirements into a structured, actionable blueprint for development. The primary focus of modelling is to clearly define the software architecture, data flow, user interaction, and system behaviour using various UML diagrams. The modelling process for the Malicious URL Detection System using Ensemble Learning Techniques includes both high-level architectural decisions and detailed behavioural and structural modeling using the Unified Modeling Language (UML).

Core Components of Malicious URL Detection System

The system comprises several core components that collaboratively work to identify and classify URLs as legitimate or phishing:

Input Interface

- Accepts URLs for classification.
- Acts as the entry point for both user-provided and dataset-based inputs.

Feature Extraction Module

- Extracts key characteristics from each URL such as:
  - Presence of IP address
  - URL length
  - Use of favicon
  - Number of subdomains
  - Use of HTTPS, '@' symbol, redirection, etc.
- Converts these characteristics into a numerical list of 30 features:
  - 1 → Feature exists
  - 0 → Not applicable
  - -1 → Feature does not exist

Data Preprocessing

- Ensures all extracted feature vectors are cleaned, standardized, and formatted for classification.
- Handles missing values and normalizes input features if necessary.

Classifier Models

- A set of machine learning algorithms trained on labeled datasets to detect phishing patterns:
  - K-Nearest Neighbors (KNN)
  - Kernel Support Vector Machine (SVM)
  - Decision Tree
  - Random Forest
- These classifiers work independently or in an ensemble configuration to improve detection accuracy.

Ensemble Learning Engine

- Combines predictions from multiple classifiers to generate a final decision.
- Uses majority voting or weighted strategies for enhanced performance.

Evaluation and Metrics Module

- Assesses model performance using:
  - Accuracy
  - Precision
  - Recall
  - F1-score
- Helps in comparing and selecting the best-performing model for deployment.

Output Module

- Displays classification results to the user: "Legitimate" or "Phishing".
- May also include risk scores or prediction confidence levels.

System Monitoring (Optional Extension)

- Can be integrated to track real-time URL inputs and detection performance.
- Useful for future updates or retraining of models with newer phishing techniques

5. 2 Analysis and Detection

Analysis methods and machine learning classifiers to provide accurate detection.

1. Feature-Based Classification

The core of the system lies in analyzing **structural and lexical features** extracted from URLs.

Features are binary or categorical, focusing on traits commonly associated with phishing (e.g., use of '@', long subdomain chains, absence of SSL).

2. Machine Learning Classification

Each extracted feature set is fed into the following classifiers:

K-Nearest Neighbors (KNN): Identifies the class of a URL based on the most common class among its closest data points.

Kernel Support Vector Machine (SVM): Separates phishing and legitimate URLs using a kernel trick to draw non-linear boundaries.

Decision Tree: Constructs rules in a tree structure based on features that maximize information gain.

Random Forest: A collection of decision trees; reduces overfitting and improves prediction accuracy.

2. Ensemble Learning Strategy

Combines the outputs from individual classifiers to improve robustness.

Utilizes a voting mechanism (majority or weighted) to make the final classification.

4. Evaluation and Metrics

Model performance is evaluated using standard classification metrics:

Accuracy: Overall correctness of the model.

Precision: Proportion of true positives among predicted positives.

Recall: Proportion of true positives identified correctly.

F1 Score: Harmonic mean of precision and recall.

These metrics ensure the selected models are both **effective** and **generalizable**.

5. Prediction Output

After classification, the system outputs whether a given URL is **"Phishing"** or **"Legitimate"**.

Can optionally include a **confidence score** for user or admin interpretation.

5.3 System Architecture Overview

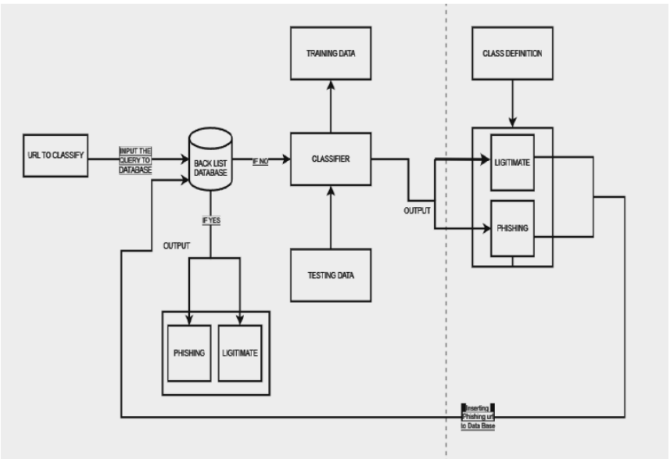


Fig 5.1 Workflow of Detection

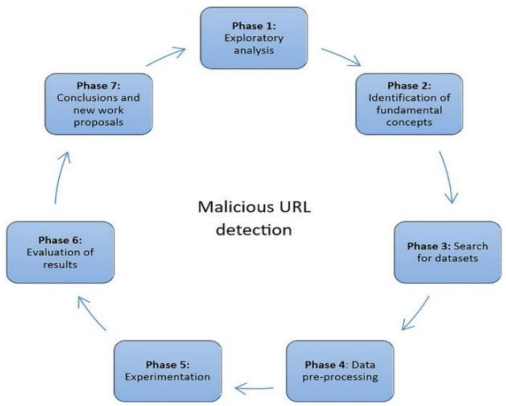


Fig 5.2 Workflow of malicious url detection

6. PROJECT IMPLEMENTATION

To implement URL classification as phishing or legitimate, we begin by selecting a dataset from Kaggle, which contains 30 features and is evenly distributed between phishing and legitimate URLs. The dataset is split into 75% for training and 25% for testing using the "train-test split" method. Preprocessing involves standardizing the features to ensure they are within a consistent range. Feature extraction is done using Python libraries such as who is, requests, Beautiful Soup, and others, to gather essential information like IP address, URL length, domain, subdomains, and the presence of a favicon. This data is then converted into a list format and fed into various classifiers, including KNN, Kernel SVM, Decision Tree, and Random Forest. These models are trained to classify URLs accurately, ultimately enabling the system to detect phishing URLs effectively.

OUTPUT

Gradient Boosting
Input URL – http://www.gaup.{BLOCKED}of.com
Algorithm – Gradient Boosting Algorithm
Expected outcome – Phishing

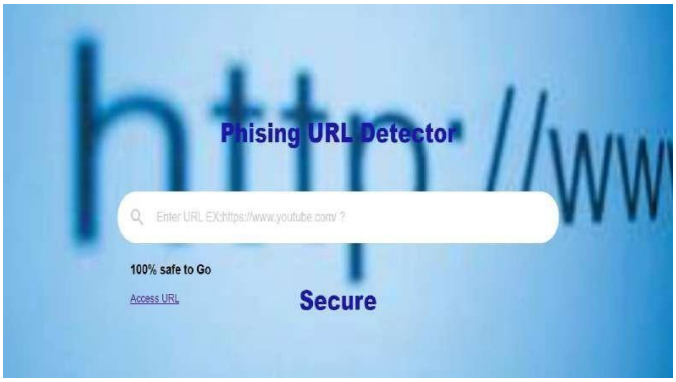
Gradient Boosting

Input URL - https://www.youtube.com

Algorithm – Gradient Boosting

Algorithm Expected outcome – Legitimate

Obtained – Legitimate



7 CONCLUSION

In conclusion, of fortfying the digital landscape publishing the url detection. The demonstration of phishing is turning into an advanced danger to this quickly developing universe of innovation. Today, every nation is focusing on cashless exchanges, business online, tickets that are paperless and so on to update with the growing world. Yet phishing is turning into an impediment to this advancement. Individuals are not feeling web is dependable now. It is conceivable to utilize AI to get information and assemble extraordinary information items. The project means to investigate this region by indicating an utilization instance of recognizing phishing sites utilizing ML. It aimed to build a phishing detection mechanism using machine learning tools and techniques which is efficient, accurate and cost effective. The project was carried out in Anaconda IDE and was written in Python. The proposed method used four machine learning classifiers to achieve this and a comparative study of the four algorithms was made. A good accuracy score was also achieved. The four algorithms used are K- Nearest neighbor, Kernel Support Vector Machine, Decision Tree and Random Forest Classifier. All the four classifiers gave promising results with the best being Random Forest Classifier with an accuracy score of 96.82%. The accuracy score might vary while using other datasets and other algorithms might provide better accuracy than random forest classifier. Random forest classifier is an ensemble classifier and hence the high accuracy. This model can be deployed in real time to detect the URLs as phishing or legitimate

8. REFERENCES

1. S. Alrefaai, G. Özdemir, A. Mohamed, Detecting Phishing Websites Using Machine Learning, in Proceedings of the 2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, Turkey (2022)

2. M. D. Bhagwat, P. H. Patil and T. S. Vishawanath, A Methodical Overview on Detection, Identification and Proactive Prevention of Phishing Websites, in Proceedings of the 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), Tirunelveli, India (2021)

3. P. Bhavani, Amba, Chalamala, Madhumitha, Likhitha, Sree Sai, C. P. Sai, Intl.J. Appl. Res. Tech 8, 2511 (2022)

4. B. B. Gupta, K. Yadav, I. Razzak, K. Psannis, A. Castiglione, X. Chang, Comp. Communi. 175 (2021)

5.S. Parekh, D. Parikh, S. Kotak, and S. Sankhe. A new method for detection of phishing websites: Url detection. In 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), pages 949952, 2018

6.M. M. Yadollahi, F. Shoeleh, E. Serkani, A. Madani, and H. Gharaee. An adaptive machine learning based approach for phishing detection using hybrid features. In 2019 5th International Conference on Web Research (ICWR), pages 281286, 2019.

7.P. Yang, G. Zhao, and P. Zeng. Phishing website detection based on multidimensional features driven by deep learning. IEEE Access, 7:1519615209, 2019

8.Muhammet Baykara and Zahit Gurel. Detection of phishing attacks. pages 15, 03 2018

9.E. Zhu, Y. Chen, C. Ye, X. Li, and F. Liu. Ofs-nn: An elective phishing websites detection model based on optimal feature selection and neural network. IEEE Access, 7:7327173284, 2019

10.E.Poornima, N.Kasiviswanath, & C.Shoba Bindu Secured Data Sharing in Groups using Attribute-Based Broadcast Encryption in Hybrid Cloud,in Proceedings of the Emerging Trends in Expert Applications and Security. Advances in Intelligent Systems and Computing, Springer, Singapore, 841 (2019)

11.D. K. Mondal, B. C. Singh, H. Hu, S. Biswas, Z. Alom, M. A. Azim, J. Inform. Secu. Appl 62 (2021)

12.H. Shirazi, K. Haefner, and I. Ray. Fresh-phish: A framework for auto- detection of phishing websites. In 2017 IEEE International Conference on Information Reuse and Integration (IRI), pages 137143, 2017

13.A. J. Park, R. N. Quadari, and H. H. Tsang. Phishing website detection framework through web scraping and data mining. In 2017 8th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), pages 680684, 2017

14.S. Haruta, H. Asahina, and I. Sasase. Visual similarity-based phishing detection scheme using image and css with target website nder. In GLOBECOM 2017- 2017 IEEE Global Communications Conference, pages 16, 2017