Phishing Website Detection Tool

June 15, 2025

Contents

1 Introduction

Phishing represents a widespread and insidious form of cyberattack, where malicious actors craft deceptive strategies to trick users into divulging sensitive personal or financial information. These attacks typically employ fraudulent websites that mimic legitimate services, often beginning their deception with cleverly disguised URLs that are difficult to distinguish from authentic ones at a glance. The challenge lies in reliably identifying such phishing attempts before any harm occurs.

This project confronts this cybersecurity threat by developing an intelligent desktop application that harnesses the power of machine learning to detect phishing websites. By analyzing URL characteristics through an advanced supervised learning model, the system classifies URLs as either phishing or legitimate in real time. This approach aims to empower users with a lightweight, accessible tool that enhances their online safety without requiring specialized technical knowledge.

2 Project Objectives

The primary goals of this project are outlined as follows:

- Design and implement a supervised machine learning model capable of accurately classifying website URLs into phishing or legitimate categories.
- Develop a lightweight, user-friendly desktop application with a graphical user interface (GUI) that enables real-time phishing detection.
- Package the application as a standalone Windows executable to facilitate easy distribution and offline usage, enhancing accessibility and convenience.

These objectives collectively seek to create a practical cybersecurity solution that balances technical sophistication with usability.

3 Technologies and Frameworks

Component	**Tool / Technology**
Programming Language	Python 3.12
Machine Learning	Scikit-learn (Random Forest Classifier)
Data Manipulation	Pandas
Feature Extraction	urllib, TLDExtract
GUI Development	Tkinter
Deployment	PyInstaller
Dataset Format	CSV (advanced_phishing_dataset.csv)

The project leverages widely-adopted Python libraries and tools to ensure robustness, maintainability, and ease of deployment.

4 Dataset Summary

The dataset utilized in this study comprises a comprehensive collection of URL records, each labeled distinctly as either “phishing” or “legitimate”. To enhance model performance, extensive feature engineering was performed to extract over thirteen unique attributes from every URL. These features encapsulate crucial indicators often associated with phishing, such as:

- URL length and hostname length, reflecting complexity and potential obfuscation.
- Number of subdomains and dot counts, which may reveal suspicious domain structures.
- Presence of special characters including `@`, `-`, and `//`, frequently used to mislead users.
- Detection of phishing-related keywords like “login”, “secure”, “verify”, and “update”.
- Use of IP addresses in URLs instead of domain names.

This multi-faceted feature set provides the model with nuanced signals to differentiate phishing sites effectively.

5 Methodology

The core methodology centers on a supervised machine learning paradigm, employing a Random Forest Classifier to leverage ensemble learning’s strengths in handling feature interactions and reducing overfitting.

5.1 Feature Set

The model ingests a carefully selected set of features extracted dynamically from each URL:

- Length of the URL and hostname.
- Usage of IP addresses in URLs.
- Counts of special characters such as “@”, “-”, and “//”.

- Number of subdomains and dots present.
- Presence of phishing-indicative keywords including “login”, “secure”, “verify”, and “update”.

5.2 Evaluation Metrics

To rigorously assess the classifier’s performance, the following metrics are employed:

Metric	**Description**
Accuracy	Overall correctness of the model’s predictions.
Precision	Proportion of predicted phishing sites that are truly phishing.
Recall	Ability to identify all actual phishing sites.
F1-Score	Harmonic mean of precision and recall, balancing both.
Confusion Matrix	Visualization of true vs. predicted classifications.

The dataset is split into training and testing subsets following an 80-20 ratio, ensuring the model is both well-trained and generalizes effectively to unseen data.

6 Application Workflow

The application's operational flow is designed to be intuitive yet robust, comprising the following steps:

1. The user inputs a website URL via the graphical user interface.
2. The application extracts the relevant features dynamically from the entered URL.
3. These features are fed into the pre-trained Random Forest machine learning model.
4. The model generates a classification prediction in real-time.
5. The application presents the classification result clearly to the user:
    - **Legitimate Website**
    - **Phishing Website**

This workflow ensures a seamless user experience, combining technical sophistication with simplicity.

7 Output and User Interface

The graphical interface emphasizes minimalism and usability. Users can enter any URL and instantly receive feedback on its safety status without needing technical expertise. The clear visual indicators and straightforward layout enhance accessibility, making the tool practical for everyday users concerned about online security.

8 Conclusion

This project validates the effectiveness of machine learning techniques, particularly Random Forest classifiers, in detecting phishing websites through in-depth URL analysis alone. The resulting desktop application is a robust, efficient, and offline-capable tool that can significantly bolster personal or organizational cybersecurity defenses. Looking forward, integrating this system with web browsers or extending its capabilities to recognize image-based phishing attempts could further enhance protection against evolving threats.

9 Acknowledgements

The development of this project was made possible through participation in a Cybersecurity and Ethical Hacking Internship Program. The author gratefully acknowledges the invaluable mentorship, coordination, and peer support that greatly contributed to the project's success.

Submitted by

D. Sai Srinivas Reddy

B.Tech –Computer Science and Engineering

Vignan's LARA Institute of Technology and Science

June 2025