أكاديمية كاوست
KAUST ACADEMY

جامعة الملك عبدالله للعلوم والتقنية
King Abdullah University of Science and Technology

Phish Me If You Can

Detecting Phishing Websites Using Machine Learning

Abstract

Phishing attacks remain one of the most common and effective methods used by cybercriminals to compromise organizational security. To address this ongoing threat, we have developed a browser extension that uses machine learning techniques to detect phishing websites in real-time. The extension analyzes URLs and website content in real-time to warn users about potentially harmful sites. Our system integrates machine learning models trained on specific features extracted from URLs and content, offering a high accuracy rate in distinguishing between legitimate and phishing sites. Additionally, the system includes a SOC (Security Operations Center) integration feature that provides organizations with detailed logs to identify at-risk users. This helps in focusing training efforts on those who need it most, thereby strengthening overall organizational security.

Written by:

Abdulrahman al-Shaabi

Abdullah al-Mashharawi

Muath Khalid

Supervised by:

Prof. Mudassar Aslam

Date:

31 August 2024

SITE
الشركة السعودية لتقنية المعلومات
Saudi Information Technology Company

الهيئة الوطنية للأمن السيبراني
National Cybersecurity Authority

1. Introduction and Problem Statement

Phishing attacks continue to be a significant threat in the digital landscape, with cybercriminals constantly refining their tactics to deceive users and steal sensitive information. Despite advancements in cybersecurity, phishing remains a pervasive issue due to the rapid creation of new phishing sites and the difficulty in distinguishing them from legitimate ones. A recent study by Proofpoint in 2024 revealed that nearly 70% of employees are likely to click on suspicious links, highlighting the need for more effective prevention tools.

To address this challenge, we developed a browser extension that provides real-time protection against phishing attacks. This tool not only detects phishing attempts but also plays a crucial role in enhancing an organization's security posture by identifying users who are most vulnerable to such attacks. The system is designed to integrate with existing SOCs, enabling targeted training for specific users that can significantly reduce the risk of successful phishing attempts.

2. Design and Implementation

2.1 System Architecture:

The system architecture is centered around a machine learning model that classifies websites as either legitimate or phishing. The architecture includes the following components:

2.1.1 Feature Extraction:

This module extracts critical features from both the URL and the HTML content of a website. These features include URL structure, suspicious patterns, the presence of certain keywords commonly associated with phishing, and other relevant indicators. By focusing on these elements, the feature extraction module ensures that the most telling characteristics of phishing sites are captured.

2.1.2 Classification:

The classification module utilizes a Random Forest classifier, known for its robustness and ability to handle a wide variety of features. This model processes the extracted features in real-time, delivering immediate feedback to users as they navigate the web. The classifier's effectiveness in handling large datasets and its interpretability make it ideal for this application.

2.1.3 User Interface and Feedback:

The user interface of the browser extension allows users to receive real-time alerts, interact with the system, and provide feedback on the model's performance. The feedback mechanism is crucial for retraining the model, enabling continuous improvement in its phishing detection accuracy.

2.2 Implementation:

The system's backend was developed using Python, with the machine learning model built using libraries such as scikit-learn. The frontend browser extension was created using HTML, CSS, and JavaScript, with RESTful APIs facilitating communication between the frontend and backend, all managed within a Flask framework.

2.2.1 Real-time Detection:

The browser extension is designed to analyze each website as it loads, using the trained model to provide immediate warnings if a phishing attempt is detected. The system allows users to report false positives or negatives, contributing to the ongoing improvement of the model.

2.2.2 SOC Integration:

The system generates detailed logs of user interactions with detected phishing websites, which can be integrated with Security Operations Centers (SOCs). These logs offer critical data, enabling organizations to identify at-risk users who frequently engage with phishing content. By focusing security training efforts on these individuals, organizations can significantly enhance their overall security posture. The targeted approach ensures that training resources are effectively allocated, addressing the most vulnerable points in the organization's defense.

3. Results and Discussion

3.1 System Testing:

The system was rigorously tested on a comprehensive dataset containing thousands of legitimate and phishing websites. The Random Forest classifier demonstrated high accuracy, consistently distinguishing between phishing and legitimate sites based on the extracted features.

3.2 SOC Integration and Impact:

The SOC integration feature proved invaluable, providing organizations with actionable insights into phishing trends and user behavior. By identifying users who are most frequently exposed to phishing attempts, organizations can tailor their training programs to address specific vulnerabilities. This approach leads to a more informed and security-conscious workforce, significantly reducing the likelihood of successful phishing attacks.

3.3 Model Performance:

The model achieved a high level of precision and recall, minimizing false positives while ensuring that potential threats were promptly identified. User feedback was instrumental in refining the model, leading to incremental improvements in its detection capabilities.

Classification Report

	precision	recall	f1-score	support
Phishing	0.967793	0.947819	0.957702	2568.0
Legitimate	0.947492	0.967587	0.957434	2499.0
accuracy	0.957569	NaN	NaN	5067.0
macro avg	0.957643	0.957703	0.957568	5067.0
weighted avg	0.957781	0.957569	0.957570	5067.0
false_positive_rate	0.052181	NaN	NaN	NaN

This visual representation of the classification report highlights the precision, recall, and F1-score achieved by the model for both phishing and legitimate websites. This classification report demonstrates the effectiveness of our approach in accurately distinguishing between legitimate and phishing sites

4. Conclusion

This project offers a robust and comprehensive solution to the ongoing threat of phishing attacks. By integrating real-time detection, SOC capabilities, and a user feedback loop, we have developed a system that not only protects users immediately but also enhances organizational security through continuous improvement. The system's design and implementation underscore its effectiveness and relevance in today’s cybersecurity landscape.

4.1 Current Feasibility

The project demonstrates a high level of feasibility in its current form. The system effectively integrates real-time phishing detection with SOC capabilities, providing immediate protection against phishing attacks while also offering valuable insights for organizational security. The use of a Random Forest classifier, coupled with extensive feature extraction, allows the model to achieve high accuracy in distinguishing between legitimate and phishing URLs. The implementation of a feedback loop further enhances the model's accuracy by enabling continuous improvement based on user reports.

From a technical standpoint, the backend infrastructure, developed using Python and Flask, ensures that the system is scalable and can be deployed in various environments, including corporate networks. The browser extension, built with HTML, CSS, and JavaScript, is lightweight and user-friendly, making it accessible for non-technical users while still providing advanced security features.

The current feasibility of the system also extends to its integration with existing SOCs. The detailed logging functionality allows for seamless integration with SOCs, providing organizations with actionable data on phishing trends and user behavior. This integration supports targeted training initiatives, making the system not only a detection tool but also a proactive component of an organization's cybersecurity strategy.

4.2 The Road Ahead

Looking forward, the project has significant potential for expansion and enhancement. The primary focus for future development will be on incorporating more advanced detection techniques, such as the use of deep learning models and Natural Language Processing (NLP) to analyze website content in greater depth. These techniques could improve the system's ability to detect sophisticated phishing attempts, such as spear-phishing and other targeted attacks.

In addition, there is potential to enhance the user training modules by leveraging the detailed logs and behavioral data collected by the system. This data can be used to develop personalized training programs that address specific vulnerabilities identified in individual users or departments. By focusing on the most at-risk users, organizations can further reduce the likelihood of successful phishing attacks and strengthen their overall security posture.

Another area of future development is the potential integration with other cybersecurity tools and platforms. By creating APIs and other integration points, the system could become part of a larger ecosystem of security tools, providing a more comprehensive defense against phishing and other cyber threats.

In conclusion, while the "Phish Me If You Can" project is already a powerful tool for phishing detection and prevention, its future development holds the promise of even greater capabilities and effectiveness. The road ahead will involve not only refining the existing system but also exploring new technologies and methodologies to stay ahead of emerging threats in the ever-evolving landscape of cybersecurity.

4.3 Future Work:

The next steps for the project will involve realistic enhancements that build upon the current foundation. These include:

4.3.1 Incorporating Behavioral Analytics:

Adding a layer of behavioral analytics to the model, which can identify abnormal user behavior patterns, will improve the system's ability to detect phishing attempts that might bypass traditional URL and content analysis.

4.3.2 Expanding SOC Integration:

Enhancing the SOC integration to support real-time incident response capabilities, allowing security teams to act swiftly on identified threats.

4.3.3 User Education Tools:

Developing interactive training modules within the extension itself, which could educate users on safe browsing practices as they interact with the tool, further reducing the risk of phishing attacks.

By focusing on these enhancements, the project will continue to evolve, providing even greater protection and adaptability in the face of emerging phishing threats.

5. GitHub Page/References and Source Code

- **PhiUSIIL Phishing URL Dataset**
  A comprehensive dataset of phishing URLs available on the UCI Machine Learning Repository.
  https://archive.ics.uci.edu/dataset/967/phiusiil+phishing+url+dataset
- **OpenPhish**
  A real-time phishing threat intelligence feed providing data on active phishing sites.
  https://www.openphish.com/
- **PhishTank**
  A community-driven website where users can report, verify, and share phishing websites.
  https://phishtank.org/
- **Criminal IP Chrome Extension**
  A Chrome extension for monitoring and detecting malicious websites, including phishing URLs.
  https://www.criminalip.io/products/chrome-extension
- **Top 1M Websites (Kaggle)**
  A dataset listing the top 1 million most visited websites, useful for comparing phishing sites against popular domains.
  https://www.kaggle.com/datasets/cheedcheed/top1m
- **Top 1 Websites (Kaggle)**
  A dataset focusing on the most visited websites, useful for understanding trends in legitimate site traffic.
  https://www.kaggle.com/datasets/cheedcheed/top1

Please visit our GitHub repository to access the source code

URL: github.com/PhishMeIfYouCan