A Proactive Approach for Fake Website Detection Using Machine Learning

N. Mounika(ASST.PROF)
Computer science&Engineering
Sasi Institute of Technology &
Engineering,Tadepalligudem
Mounika07@sasi.ac.in

G.P. Amrutha
Computer science&Engineering
Sasi Institute of Technology &
Engineering,Tadepalligudem
amrutha.gampala@sasi.ac.in

V. Krupavani
Computer science& Engineering
Sasi Institute of Technology &
Engineering,Tadepalligudem
krupavani.vallepalli@sasi.ac.in

CH. Kumar Raja
Computer science&Engineering
Sasi Institute of Technology &
Engineering,Tadepalligudem
kumarraja.chinthalapudi@sasi.ac.in

A. Raj Prakash
Computer science&Engineering
Sasi Institute of Technology &
Engineering,Tadepalligudem
prakash.ambati@sasi.ac.in

ABSTRACT:

Phishing is a cyber-attack technique that employs fraudulent websites to trick individuals into revealing sensitive information, such as passwords, usernames, and financial details. With the rising prevalence of phishing attacks, developing robust detection systems is essential to safeguard individuals and organizations. The proposed system leverages a dataset comprising features derived from website components, including URL structure, domain attributes, and content properties. In the preprocessing stage, the dataset is refined by cleaning the data and selecting pertinent features to enhance the identification of phishing websites. The system employs advanced classification models, namely XGBoost, CatBoost, and LightGBM, to accurately distinguish between legitimate and phishing websites. These gradient-boosting-based approaches enhance online security and mitigate the risks posed by phishing attacks.

Keywords: Phishing, Cybersecurity, Machine Learning, XGBoost, CatBoost, LightGBM

Introduction

Phishing is a significant online threat in which cybercriminals create deceptive websites to trick users into sharing sensitive information like usernames, passwords, and financial details. These phishing attacks often involve malicious URLs that appear to be legitimate, mimicking trusted websites to deceive individuals into revealing personal data. Phishing is one of the most common techniques attackers use to gather confidential information, making it crucial to detect and prevent such attacks as early as possible.

Phishing websites are specifically designed to resemble legitimate ones, often copying the layout, logos, and design of trusted organizations like banks, e-commerce platforms, or email providers. The goal is to make users believe they are on a genuine site and prompt them to enter their personal information, which is then harvested by the attackers. These fraudulent sites may be promoted through various methods such as fake emails, SMS messages, or social media posts, all of which include malicious links that direct unsuspecting users to phishing sites.

One of the most critical aspects of phishing attacks is the use of phishing URLs, which are links crafted to look trustworthy but lead to malicious websites. These URLs may include minor changes, such as substituting a letter or number in a familiar domain name (for example, replacing an "o" with a zero) to trick users into thinking they are accessing a legitimate site. Phishing URLs may also appear to use secure HTTPS protocols, adding to their deceptive appearance. However, the presence of HTTPS alone does not guarantee the safety of a website.

The consequences of falling victim to a phishing website can be severe. Once cybercriminals obtain sensitive information, they can use it for various malicious purposes, including financial theft, identity fraud, and unauthorized access to online accounts. Individuals may experience financial losses, while businesses can face significant disruptions, including data breaches, loss of customer trust, and reputational damage.

Detecting and blocking phishing websites is essential to minimizing these risks. One common approach is through whitelisting and blacklisting. Whitelisting involves maintaining a list of trusted websites that are known to be safe. If a user tries to access a site, the system checks whether the URL matches any entry on the whitelist. If it does, the site is allowed. Conversely, blacklisting keeps track of known malicious URLs. When a user attempts to visit a website, the system compares the URL to the blacklist, blocking it if it's recognized as harmful.

While whitelisting and blacklisting are effective to a certain extent, cybercriminals constantly create new phishing URLs, which may not immediately appear on blacklists. This dynamic nature of phishing attacks makes it challenging to rely solely on these methods, as newly created phishing sites can still slip through undetected until added to the list.

Another effective method for identifying phishing websites is heuristic analysis, which analyzes the behavior and characteristics of URLs and websites to detect anomalies that indicate potential threats. Instead of relying on a fixed list of trusted or malicious URLs, heuristic analysis looks at various factors such as the URL's structure, the use of suspicious characters, the website's domain age, and content analysis to identify phishing attempts. For instance, phishing websites often have longer and more complex URLs, as well as domain names that have been recently registered. They may also lack detailed contact information or have inconsistencies in their content, such as poor grammar or broken links.

A study by Enisa [1] highlights that phishing attacks are among the most frequent cyber threats faced by small and medium-sized enterprises (SMEs) in Europe. According to Cisco's Cybersecurity Threat Trends report [2], phishing was responsible for approximately 90% of data breaches in 2020. Additionally, 86% of organizations had at least one employee attempt to access a phishing site. As noted in [3], one of the primary reasons individuals fall victim to phishing is due to insufficient care in verifying the legitimacy of websites, as well as a lack of proper cybersecurity education.

Phishing websites are also dangerous because they can deliver malware to a user's device. When a user clicks on a phishing link, it may not only lead to a fraudulent site but could also trigger the download of malicious software. This malware can infect the user's computer, allowing attackers to gain access to personal files, monitor activities, or spread the infection to other systems within a network. For organizations, such malware infections can lead to widespread data breaches or ransomware attacks, causing significant operational and financial damage.

Preventing phishing attacks also helps organizations maintain their reputation. If a business falls victim to a phishing attack, the data compromised can include customer information, leading to loss of trust and credibility. Customers who lose confidence in a company's ability to protect their data may seek alternatives, and the organization could face legal consequences and financial penalties related to data privacy violations.

In conclusion, phishing websites are a prevalent and dangerous method used by cybercriminals to steal sensitive information. They often use deceptive URLs to lure users into entering personal details or downloading malicious software. Detecting phishing websites through whitelisting, blacklisting, and heuristic analysis is essential for minimizing the risk of phishing attacks. By preventing these attacks, individuals and organizations can protect themselves from financial losses, data breaches, and reputational harm.

Literature survey

Das Gupta, Sumitra, and colleagues explored different approaches such as blacklisting, whitelisting, heuristic analysis, and visual similarity to detect newly launched phishing websites. Their research emphasized the difficulty of real-time phishing detection. The authors created a method using the Document Object Model (DOM) to aid in identifying phishing sites, leading to higher accuracy and fewer errors from the best classifier in their evaluations [4].

SM Istiaque introduced a new approach demonstrating the potential of artificial intelligence (AI) in cybersecurity. The study used several machine learning models to show how AI can detect phishing attacks. In the study's two-step validation process, AI models were trained and tested using the KDD'99 dataset, a well-known resource in cybersecurity research [5].

R. Yaqoob focused on using custom XPATH to launch attacks for real-time price scraping on websites like Alibaba and eBay. The study revealed that bot attacks continue to be a threat, as XPATH copying techniques are not restricted, allowing hackers to scrape data from these platforms [6].

Ashit Kumar Dutta developed a system using Naive Bayes and Support Vector Machine (SVM) classifiers to detect malicious URLs. The system utilized a dataset combining Alexa Rank and PhishTank sources, employing tools like WordNet and specific seed words to distinguish between good and bad websites [7].

S. Anupam and colleagues applied four optimization methods in addition to using an SVM binary classifier to identify phishing websites. They incorporated the Grey Wolf Optimizer algorithm with a Random Forest classifier, which outperformed other techniques in terms of phishing detection accuracy [8].

Ghaleb AI-Mekhlafj and co-authors employed several machine learning algorithms, optimizing parameters to enhance phishing detection. They worked with datasets of 4898 and 6157 records, focusing on improving efficiency by using ensemble classification methods and genetic algorithms [9].

KS Swarnalatha's study highlighted that phishing websites often perfectly mimic legitimate ones in appearance and content. These websites aim to steal sensitive information like usernames and passwords. The study classified phishing attacks as social engineering attacks, where hackers exploit human trust [10].

Gururaj and colleagues used various methods, including wrapper-based feature selection, K-Nearest Neighbors (KNN), Random Forest, Decision Trees, single-class SVM, and linear classification to verify website authenticity. Their multi-technique approach provided a thorough method for phishing detection [11].

R. Nanjundappa discussed how AI and machine learning models can help web application developers address security and user experience challenges. The study highlighted the need for frameworks that simplify AI-based web app development, especially in improving security and content analysis [12].

Rashid applied machine learning algorithms, particularly combining Principal Component Analysis (PCA) with SVM, to detect phishing websites. By reducing feature dimensions, PCA helped improve the efficiency and accuracy of the SVM model in phishing detection [13].

Ankit Kumar and colleagues introduced the PhishSkape model, which distinguished between phishing and legitimate websites. The model was tested on 200 websites and proved effective in identifying phishing threats [14].

Teja evaluated multiple machine learning models on the Kaggle Phishing Website Dataset, concluding that the Random Forest Classifier (RFC) was the most accurate method for detecting phishing websites [15].

Abdul Basit developed a voting algorithm that combined Random Forest, Artificial Neural Networks (ANN), and C4.5 algorithms. This hybrid model demonstrated high predictive accuracy and a strong ROC Area score, proving its effectiveness in phishing detection [16].

Nigraha and Rahman significantly improved phishing detection performance, achieving high accuracy on their dataset. Their results indicate the potential of machine learning techniques in enhancing phishing detection [17].

S. Jagadeesan conducted a comparative study of Random Forest and two SVM models. The results showed that Random Forest offered superior performance in phishing detection, surpassing the other models [18].

Jain and Gupta review phishing detection approaches based on visual similarity, classifying them by feature types (e.g., visual, pixel-based). The survey is focused only on visual similarity methods and does not cover other techniques[19].

Sahoo et al. [20]discuss phishing detection using page URLs, presenting feature representations and machine learning algorithms. However, the paper doesn’t explore methods that integrate multiple content types like HTML or visual features.

Das et al.[21] focus on phishing detection across URLs, websites, and emails using various machine-learning methods. They discuss computation and storage challenges but don’t compare machine learning with other detection methods.

This survey provides a high-level analysis of AI-enabled phishing detection techniques, including machine learning and deep learning. However, it reviews a few papers and omits pioneering machine learning studies[22].

Dou et al. review software-based phishing detection schemes, covering taxonomy, datasets, features, and techniques. They thoroughly analyze 12 representative papers based on page content, URL, and hybrid detection methods[23].

Prakash et al. [24] propose an offline method to generate new URLs from blacklisted ones by applying URL lexical similarity heuristics. The method validates generated URLs using DNS lookup and content matching before adding them to blacklists, discarding non-existent or harmless URLs.

This approach uses probabilistic detection[25] to find near-duplicate phishing pages by combining human-verified blacklists and the shingling algorithm. It also queries search engines with content extracted from suspected phishing pages to enhance detection.

Rao and Pais [26], Their method identifies phishing web pages by comparing the fingerprints of suspicious pages with blacklisted ones using the Hamming distance. These fingerprints are based on features extracted from the source code of web pages.

This strategy detects new URLs by tracking redirections from blacklisted URLs and following phishing forms iteratively, aiming to populate blacklists quickly and effectively[27].

Cao et al. [28] use a Naïve Bayesian classifier to update user whitelists by adding login interface information (e.g., URL, DNS-IP mapping) after a user successfully logs in multiple times, ensuring secure login processes.

This method auto-updates whitelists by checking the legitimacy of pages accessed by users. The check is based on hyperlink features from the page's source code, as phishing pages often include links to legitimate sites[29].

A multilayer model is used to update whitelists, assessing the legitimacy of URLs by analyzing features, lexical signatures, and search engine rankings. Legitimate pages are typically ranked highly in search results, providing an extra layer of verification[30].

A detailed review of phishing detection strategies is provided, categorized into six techniques search-based, heuristics, black/whitelists, and visual similarity. The survey discusses the pros and cons of each method but lacks clarity on paper selection criteria and has limited machine learning coverage[31].

Author	**Technique**	**Category**	**Accuracy**	**Year**
Narmatha C. et al.	Support Vector Machine, Logistic Regression, Random Forest	Machine Learning	86.05%	2022
Subhash Ariyadas A.	XG Boost, Random Forest	Machine Learning	49.88%	2022
Jitendra Kumar	Convolutional Neural Network	Deep Learning	95.62%	2021
Abdul Afeez Wojuade	Heuristics Based Features	Heuristics Based Approach	96%	2022
Mary Isangedi Ok	Decision Tree, Random Forest	Machine Learning	78%	2022
Yuba R. Siwakoti and Danda B. Rawat	ORB Features Extraction, Random Forest	Machine Learning	92.63%	2022
Alamughaid	MHSA, Convolutional Neural Network	Deep Learning	93.05%	2022

Methodology

A)Proposed system

The proposed system aims to detect phishing websites by employing advanced machine learning techniques, specifically XGBoost, CatBoost, and LightGBM classifiers. It utilizes a diverse set of features extracted from URLs, such as their structure, length, and the presence of suspicious keywords, to evaluate the legitimacy of websites. The system builds upon existing research, such as the work by Narmatha C. et al., which achieved an accuracy of 86.05% using Support Vector Machine, Logistic Regression, and Random Forest algorithms for similar detection tasks. By training the XGBoost, CatBoost, and LightGBM models on a labeled dataset of phishing and legitimate websites, the proposed system enhances its capability to classify incoming URLs in real time, offering users a powerful tool to combat online fraud.

Additionally, the proposed system integrates these gradient-boosting-based classifiers—XGBoost, CatBoost, and LightGBM—to provide a robust and efficient approach to phishing detection, leveraging their high performance and ability to handle complex datasets. Research conducted by Mary Isangedi Ok indicates that Decision Tree and Random Forest methods achieved an accuracy of 78% in identifying fraudulent websites. By adopting XGBoost, CatBoost, and LightGBM, the proposed system not only aims to surpass these benchmarks in detection accuracy but also strives to deliver valuable insights into the classification process. This multi-model strategy strengthens the overall cybersecurity framework, significantly improving the safety and reliability of internet browsing for users.

B)System Architecture

The proposed system architecture for detecting phishing websites comprises several key components that collaborate to analyze and classify URLs effectively. Central to the architecture is a feature extraction module that collects critical data from input URLs, including their length, the presence of special characters, and other structural attributes. This module processes the incoming URL data and converts it into a structured format optimized for machine learning algorithms. The processed features are then passed to three advanced classifiers: XGBoost, CatBoost, and LightGBM. These classifiers, trained on a labeled dataset of phishing and legitimate websites, assess the URLs based on the extracted features to determine their legitimacy.

The outputs from XGBoost, CatBoost, and LightGBM are combined through an ensemble approach, boosting the overall detection accuracy. The final classification result is delivered to the user, clearly indicating whether a URL is phishing or safe. Additionally, the architecture incorporates a user interface that enables users to submit URLs for analysis and view results in real time. Designed for efficiency, the system ensures rapid responses to user queries, providing a seamless experience while bolstering cybersecurity. Overall, this architecture not only enables precise phishing detection but also emphasizes user engagement and security awareness.

C)Datasets

The dataset utilized in this study, sourced from Kaggle, comprises data on 11,430 URLs, each characterized by 87 features. It is evenly balanced, containing an equal distribution of phishing and legitimate website examples, making it well-suited for training and evaluating machine learning models for phishing detection.

D)Algorithms

i)XgBoost

XGBoost, short for eXtreme Gradient Boosting, is a powerful and widely-used machine learning algorithm designed for supervised learning tasks, particularly classification and regression. It belongs to the family of gradient boosting techniques, which build an ensemble of weak learners—typically decision trees—in a sequential manner to improve predictive performance. XGBoost stands out due to its efficiency, scalability, and ability to handle large datasets with high-dimensional features, making it an ideal choice for applications like phishing website detection. The algorithm optimizes a loss function by iteratively adding trees that correct the errors of previous ones, using gradient descent to minimize the overall error. Its key strengths include regularization to prevent overfitting, parallel processing for faster computation, and the ability to handle missing data effectively.

In the context of phishing detection, XGBoost excels by leveraging its capability to model complex, non-linear relationships within URL features, such as length, special characters, and structural patterns. It employs a sophisticated tree-boosting framework that incorporates features like weighted quantile sketching for efficient split finding and sparsity-aware algorithms to optimize performance on sparse datasets. Additionally, XGBoost offers flexibility through customizable hyperparameters, allowing fine-tuning of aspects like learning rate, tree depth, and subsample ratio to enhance accuracy and robustness. Its proven track record in machine learning competitions and real-world applications underscores its reliability, often outperforming traditional algorithms like Support Vector Machines or standalone Decision Trees, especially when combined with ensemble strategies in cybersecurity tasks.

ii)Catboost

CatBoost, short for Categorical Boosting, is an advanced gradient boosting algorithm developed by Yandex, designed to handle a wide range of machine learning tasks, including classification and regression. It builds on the principles of gradient boosting by sequentially constructing an ensemble of decision trees, where each tree corrects the errors of its predecessors. What sets CatBoost apart is its native support for categorical features, eliminating the need for extensive preprocessing like one-hot encoding, which is particularly advantageous when dealing with URL-based datasets that may include categorical attributes like domain names or protocols. This efficiency, combined with its ability to deliver high accuracy, makes CatBoost a strong candidate for phishing website detection, where diverse and complex feature sets are common.

In practical applications such as identifying phishing URLs, CatBoost leverages its robust handling of overfitting through an innovative ordered boosting technique, which reduces bias and improves generalization compared to traditional boosting methods. It also incorporates symmetric trees and oblivious decision trees, ensuring consistent and efficient feature evaluation, which enhances computational speed without sacrificing performance. CatBoost's flexibility is further augmented by its support for customizable hyperparameters, such as learning rate, depth, and L2 regularization, allowing it to adapt to the specific needs of a balanced dataset like the one with 11,430 URLs and 87 features.

iii)LightGBM

LightGBM, or Light Gradient Boosting Machine, is a high-performance machine learning framework developed by Microsoft, specifically optimized for gradient boosting tasks such as classification and regression. Built on the foundation of decision tree ensembles, LightGBM distinguishes itself with its exceptional speed and scalability, making it well-suited for processing large datasets like the one containing 11,430 URLs with 87 features. Unlike traditional boosting methods that grow trees level-wise, LightGBM employs a leaf-wise tree growth strategy, where it selects the leaf with the maximum loss reduction to split, resulting in faster training and often higher accuracy. This efficiency, combined with its ability to handle high-dimensional data, positions LightGBM as an effective tool for detecting phishing websites by analyzing complex URL features such as length, structure, and special characters.

In the context of phishing detection, LightGBM offers additional advantages through its support for histogram-based learning, which bins continuous features into discrete intervals, significantly reducing memory usage and accelerating computation. It also includes optimizations like Gradient-based One-Side Sampling (GOSS), which prioritizes instances with larger gradients, and Exclusive Feature Bundling (EFB), which groups mutually exclusive features to further enhance performance on sparse datasets. These features make LightGBM particularly adept at handling the diverse and potentially noisy attributes of URLs. With customizable hyperparameters such as learning rate, number of leaves, and boosting iterations, LightGBM can be fine-tuned to maximize detection accuracy while maintaining robustness.

Experimental Results

1. Accuracy

Conclusion

The research in phishing detection has produced a wide array of methodologies and insights aimed at enhancing the identification and mitigation of phishing attacks. Various strategies have been explored, including blacklisting, whitelisting, heuristic analysis, visual similarity, and machine learning techniques. In this study, the proposed system leverages advanced classifiers—XGBoost, CatBoost, and LightGBM—to effectively detect phishing websites, utilizing a balanced dataset of 11,430 URLs described by 87 features sourced from Kaggle. These gradient-boosting-based algorithms have demonstrated their capability to analyze intricate URL attributes, such as structure, length, and special characters, outperforming traditional approaches like Support Vector Machines (SVM) and Random Forest in terms of efficiency and accuracy.

The proposed system emphasizes the integration of multiple detection techniques to improve performance and minimize false positives. By combining the strengths of XGBoost, CatBoost, and LightGBM through an ensemble approach, the system optimizes feature evaluation and classification, building on findings from prior studies that highlight the value of ensemble methods and feature selection. The architecture also prioritizes real-time detection, addressing the evolving tactics of cybercriminals through rapid URL

Accuracy is a measure of the overall correctness of a classification model, representing the proportion of true results (both true positives and true negatives) among the total cases examined. It indicates how well the model performs across all classes.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

2. Precision

Precision measures the accuracy of positive predictions made by the model. It represents the proportion of true positive results to the total predicted positives. High precision indicates that the model has a low false positive rate.

$$\text{Precision} = \frac{\textit{True Positive}}{\textit{True Positive} + \textit{False Positive}}$$

3. Recall

Recall, also known as sensitivity, measures the ability of a model to identify all relevant instances within the positive class. It represents the proportion of true positives to the total actual positives. High recall indicates that the model effectively captures positive **cases.**

4. F1 Score

$$\textit{Recall} = \frac{TP}{TP + FN}$$

$$\textbf{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

The F1 score is the harmonic mean of precision and recall, providing a single metric that balances the trade-off between the two. It is particularly useful when the class distribution is imbalanced, as it reflects the performance of the model in scenarios where both false positives and false negatives are critical.

analysis and user-friendly feedback via an interactive interface. Collectively, this research underscores the growing complexity of phishing threats and reinforces the need for sophisticated, scalable solutions, with the proposed system offering a robust framework to enhance online security and protect users from these persistent cyber threats.

References

[1] *Cybersecurity for SMEs—Challenges and Recommendations*, 2021, [online] Available: https://www.enisa.europa.eu/publications/enisa-report-cybersecurity-for-smes.

[2] Cyber Security Threat Trends: Phishing Crypto Top the List, 2021, [online] Available: https://umbrella.cisco.com/info/2021-cybersecurity-threat-trends-phishing-crypto-top-the-list.

[3] M. Alsharnouby, F. Alaca, and S. Chiasson, "Why phishing still works: User strategies for combating phishing attacks", *Int. J. Hum.-Comput. Stud.*, vol. 82, pp. 69-82, Oct. 2015.

[4] K. T. Das Gupta, H. Shahriar, D. Alqahtani, I. H. Alsalman, and D. Alsalman, "Modeling Hybrid Feature-Based Phishing Websites Detection Using Machine Learning Techniques," 2022.

[5] S. M. Istiaque, M. T. Tahmid, A. I. Khan, Z. A. Hassan, and S. Waheed, "Man-made intellectual ability Based Organization security: Two-Step Suitability Test," 2021.

[6] R. Yaqoob, Sanaa, M. Haris, Samadyar, and M. A. Shah, "The Worth Scratching Bot Risk on Web Business Store Using Custom XPATH Strategy," 2021.

[7] A. K. Dutta, "Detecting phishing websites using machine learning technique," 2021.

[8] S. Anupam and A. K. Kar, "Phishing website detection using support vector machines and nature-inspired optimization algorithms," 2021.

[9] B. A. Mohammed and Z. G. Al-Mekhlafi, "Optimized Stacking Ensemble Model to Detect Phishing Websites," 2021.

[10] K. S. Swarnalatha, K. C. Ramchandra, K. Ansari, L. Ojha, and S. S. Sharma, "Steady Peril Information Block Phishing Attacks," 2021.

[11] G. H. Lokesh and G. BoreGowda, "Phishing website detection based on effective machine learning approach," 2021.

[12] R. Nanjundappa et al., "AWAF: man-made insight -enabled Web Things Composing Framework," 2020.

[13] J. Rashid, T. Mahmood, M. W. Nisar, and T. Nazir, "Phishing Detection Using Machine Learning Technique," 2020.

[14] A. K. Jain, "PhishSKaPe: A Content-based Approach to Escape Phishing Attacks," 2020.

[15] C. S. B. Teja, T. S. S. Sasank, and Y. J. S. Reddy, "Phishing website detection using different machine learning techniques," 2020.

[16] A. Basit, M. Zafar, A. R. Javed, and Z. Jalil, "A Novel Ensemble Machine Learning Method to Detect Phishing Attack," 2020.

[17] A. F. Nugraha and L. Rahman, "Meta-Algorithms for Improving Classification Performance in the Web-phishing Detection Process," 2019.

[18] S. Jagadeesan, A. Chaturvedi, and S. Kumar, "URL phishing analysis using random forest," 2018.

[19] A. K. Jain and B. B. Gupta, "Phishing detection: Analysis of visual similarity-based approaches", Secur. Commun. Netw., vol. 2017, pp. 1-20, Jan. 2017.

[20]D. Sahoo, C. Liu and S. C. H. Hoi, "Malicious URL detection using machine learning: A survey", *arXiv:1701.07179*, 2017.

[21]A. Das, S. Baki, A. El Aassal, R. Verma and A. Dunbar, "SoK: A comprehensive reexamination of phishing research from the security perspective", *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 671-708, 1st Quart. 2020.

[22]A. Basit, M. Zafar, X. Liu, A. R. Javed, Z. Jalil, and K. Kifayat, "A comprehensive survey of AI-enabled phishing attacks detection techniques", *Telecommun. Syst.*, vol. 76, no. 1, pp. 139-154, Jan. 2021.

[23]Z. Dou, I. Khalil, A. Khreishah, A. Al-Fuqaha, and M. Guizani, "Systematization of knowledge (SoK): A systematic review of software-based web phishing detection", *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2797-2819, 4th Quart. 2017.

[24] P. Prakash, M. Kumar, R. R. Kompella and M. Gupta, "PhishNet: Predictive blacklisting to detect phishing attacks", Proc. IEEE INFOCOM, pp. 1-5, Mar. 2010.

[25]G. Xiang, B. A. Pendleton, J. Hong and C. P. Rose, "A hierarchical adaptive probabilistic approach for zero hour phish detection" in Computer Security—ESORICS, Berlin, Germany: Springer, vol. 6345, pp. 268-285, 2010.

[26]R. S. Rao and A. R. Pais, "An enhanced blacklist method to detect phishing websites" in Information Systems Security, Berlin, Germany: Springer, vol. 10717, pp. 323-333, 2017.

[27]L.-H. Lee, K.-C. Lee, H.-H. Chen and Y.-H. Tseng, "POSTER: Proactive blacklist update for anti-phishing", *Proc. ACMSIGSAC Conf. Comput. Commun. Secur.*, pp. 1448-1450, Nov. 2014.

[28].Y. Cao, W. Han, and Y. Le, "Anti-phishing based on automated individual white-list", *Proc. 4th ACM Workshop Digit. Identity Manag.*, pp. 51-60, Oct. 2008.

[29]A. K. Jain and B. B. Gupta, "A novel approach to protect against phishing attacks at client side using auto-updated white-list", *EURASIP J. Inf. Secure.*, vol. 2016, no. 1, pp. 1-11, Dec. 2016.

[30]G. Sonowal and K. S. Kuppusamy, "PhiDMA—A phishing detection model the with a multi-filter approach", *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 32, no. 1, pp. 99-112, 2020.

[31] G. Varshney, M. Misra, and P. K. Atrey, "A survey and classification of web phishing detection schemes", *Secur. Commun. Netw.*, vol. 9, no. 18, pp. 6266-6284, Dec. 2016.