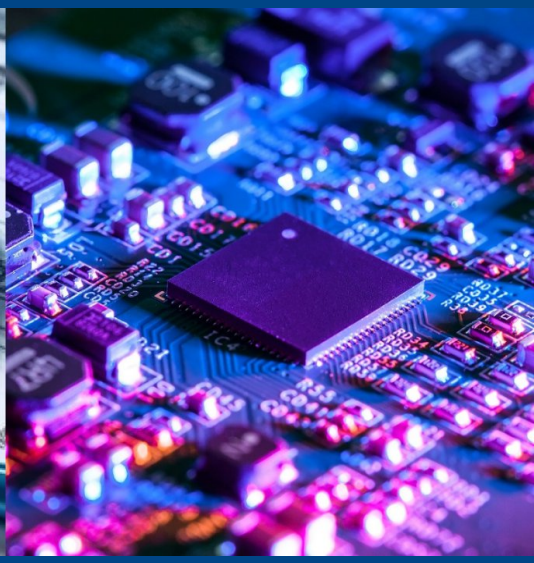
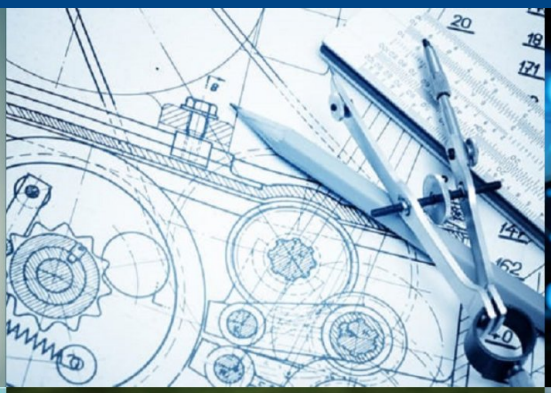
ISSN: 2582-7219

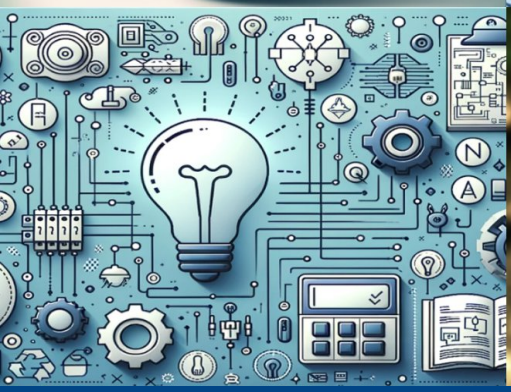




International Journal of Multidisciplinary Research in Science, Engineering and Technology

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.206 **Volume 9, Issue 4, April 2026**

© 2026 IJMRSET | Volume 9, Issue 4, April 2026 | DOI:10.15680/IJMRSET.2026.0904255

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Secure Click: A Machine Learning-Powered Browser Extension for Real-Time Phishing URL Detection

Dr. A.Vidhya [1], Mohamed Waseem S[2], Imrankhan K[3]

Assistant Professor, B.Sc. Computer Science, B. S. Abdur Rahman Crescent Institute of Science and Technology Vandalur, Chennai, Tamil Nadu, India[1]

Student, B.Sc. Computer Science, Department of Computer Application, B. S. Abdur Rahman Crescent Institute of Science and Technology, Vandalur, Chennai, Tamil Nadu, India[2]

Student B.Sc. Computer Science, Department of Computer Application, B. S. Abdur Rahman Crescent Institute of Science and Technology, Vandalur, Chennai, Tamil Nadu, India[3]

ABSTRACT: Phishing is one of the most serious and growing cybersecurity threats in today's digital enviroment. Attackers use fake websites and deceptive URLs to steal sensitive informations such as usernames, passwords, banking credentials, and personal details. Traditional phishing detection methods, especailly blacklist-based approaches, are limited because they often fail to detect newly generated phishing websites that has not yet been reported. To address this issue, this paper proposes SecureClick, a machine learning- based phishing URL detection system integrated into a browser extension for real-time user protection. The proposed system extracts important lexical and structural features from URLs, including URL length, number of dots, number of subdomains, HTTPS status, suspicious symbols, and domain-related indicators. These features are used to train machine learning algorithms such as Logistic Regression, Decision Tree, Random Forest, and Support Vector Machine to classify URLs as either phishing or legitimate. The trained model is then integrated into a browser extension that monitors visited URLs and instantly warns users when a suspicious website is detected. Our results shows that Random Forest algorithm achieved the best performance with an accuracy of 97.2%, demonstrating that machine learning can plays a significant role in real-time browser-based cybersecurity applications.

KEYWORDS: Phishing Detection, Browser Extension, Machine Learning, Cybersecurity, URL Classification, Secure Browsing

I. INTRODUCTION

The internet has became an essential part of daily life for communication, banking, shopping, education, and business transactions. However, the rapid growth of digital platforms has also increased alot of cyber threats that target internet users. Among these threats, phishing is one of the most common and dangerous attacks. In phishing, attackers create fake websites or malicious URLs that closely resemble legitimate ones inorder to deceive users into revealing confidential data.

Phishing URLs are often designed to look trustworthy by using domain names similar to real websites, adding misleading subdomains, or inserting characters that resembles legitimate brands. Many users are unable to identify such deceptive links, specially when they appear convincing. Since user's mainly access websites through browsers, implementing phishing detection directly within the browser can provide immediate protection.

Conventional security mechanisms such as blacklists are only effective against already known malicious URLs. They are not capable of identifying zero-day phishing attacks or newly created malicious domains. Therefore, there is a strong need for an intelligent and real-time phishing detection mechanism. Machine learning offers a powerful solution by learning patterns from phishing and legitimate URLs and predicting the nature of a URL based on it's extracted features. This paper presents SecureClick, a browser extension that uses machine learning to detect phishing URLs and alert users during web browsing.

In this paper, we describe the design, implementation and evaluation of SecureClick in detail. We present experimental

© 2026 IJMRSET | Volume 9, Issue 4, April 2026 | DOI:10.15680/IJMRSET.2026.0904255

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

results comparing four different classification algorithms and discuss the trade-offs between them. Our aim is to demonstrate that an effective, real-time phishing detection system can be build as a lightweight browser extension without sacrificing accuracy or user experience.

II. PROBLEM STATEMENT

Phishing attacks continue to cause major financial loss, identity theft, and privacy violations world wide. Existing browser protection systems depend heavily on blacklist databases, which are ineffective against newly launched phishing websites that has not yet been reported. As a result, users remains vulnerable to malicious links during the early stage of an attack. There is a need for a browser- based detection system that can intelligently analyze URL's in real time and warn users before they interact with fraudulent websites.

The core limitation of todays most widely deployed phishing defenses is that they are inherently reactive. A phishing site that was registered and launched this morning may not appear on any blacklist until it has already victimized hundreds or thousands of users. This temporal gap is the fundamental problem that SecureClick aims to address.

III. OBJECTIVES

The work described in this paper is guided by the following objectives:

1. To design a phishing URL detection framework using machine learning algorithms.
2. To extract and analyze important URL-based features for phishing classification.
3. To train and compare multiple machine learning algorithms for phishing detection.
4. To develop a browser extension that integrates the trained phishing detection model for real time use.
5. To provide real-time alerts to users when suspicious URLs are been detected.
6. To evaluate the complete system and report it's performance on held-out test data.

IV. LITERATURE REVIEW

Phishing detection has been widely studied using blacklist methods, heuristic analysis, visual similarity techniques, and machine learning approaches. Blacklist-based techniques compare a URL against a stored list of known phishing websites. Although they are simple and fast, they fail to detect newly generated phishing URLs.Heuristic approaches examine suspicious URL patterns such as unusually long URLs, use of IP addresses, excessive subdomains, or the presence of special symbols like '@' and '-'. These methods are more flexible then blacklists but may generate false positives for legitimate websites with uncommon structures.

Machine learning based phishing detection has shown very promising results in recent years. Algorithms such as Naive Bayes, Logistic Regression, Decision Tree, Random Forest, and Support Vector Machine have been applied to classify URLs using lexical, structural, and host-based features. Among these, Random Forest and Support Vector Machine often achieves higher accuracy. However, many existing studies focus only on the classification model and not on real-time practical implementation.

This paper address that gap by integrating a machine learning model into a browser extension that can actively protect users while browsing.

Approach	Real-Time	ML-Based	Browser	Zero-Day
Blacklist (Traditional)	Yes	No	Yes	No
Heuristic Rules	Yes	No	Partial	Partial
ML Standalone Model	No	Yes	No	Yes

© 2026 IJMRSET | Volume 9, Issue 4, April 2026 | DOI:10.15680/IJMRSET.2026.0904255

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

SecureClick (Proposed)	**Yes**	**Yes**	**Yes**	**Yes**

Table I. Comparison of SecureClick against representative phishing detection approaches

V. PROPOSED SYSTEM

The proposed system, SecureClick, is a phishing URL detection solution implemented as a browser extension. The system captures the active URL from the browser, extracts relevant features, preprocesses the feature values, and sends them to a trained machine learning model. Based on the model's output, the URL is classified as phishing or legitimate. If the URL is found to be suspicious, the extension generates an alert message or warning popup to the user. If the URL is safe, the browser continues it's normal operation. This design enables real-time detection and prevents users from unknowingly visiting dangerous websites. Fig. 1 illustrates the complete system workflow.

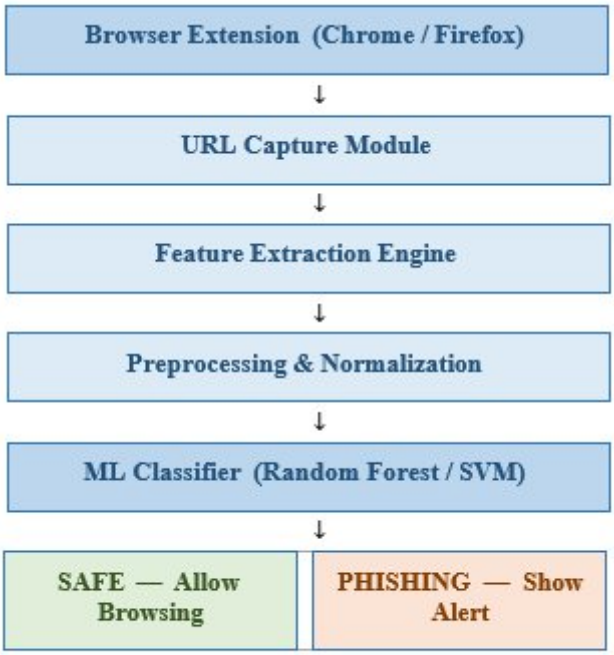


Fig. 1. SecureClick system architecture: end-to-end flow from browser URL capture to user alert or safe browsing confirmation.

VI. SYSTEM ARCHITECTURE

The architecture of SecureClick begins with the browser extension monitoring the active tab. Whenever a user opens or enters a URL, the extension captures the address and forwards it to the feature extraction unit. The feature extraction module derives lexical and structural characteristics from the URL, such as it's length, number of dots, suspicious symbols, subdomain count, protocol usage, and domain properties.

The browser extension is build using the WebExtensions API, which is supported by Chromium-based browsers such as Chrome, Edge and Brave. The extension registers a listener on the browser's webNavigation.onBeforeNavigate event, which fires before the browser begins loading a new page. This timing is crucial because by intercepting the navigation at this point, the extension can display a warning before any content from the phishing site is loaded or executed.The extracted feature set is then preprocessed and passed to the trained machine

© 2026 IJMRSET | Volume 9, Issue 4, April 2026 | DOI:10.15680/IJMRSET.2026.0904255

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

learning classifier. The classifier predicts whether the URL is phishing or legitimate. Based on this prediction, the system either displays a warning message or allows the user to continue browsing safely.

VII. DATASET DESCRIPTION

The dataset used for this work consists of phishing and legitimate URLs collected from publicly available cybersecurity repositories and safe web sources. Each URL is labeled as either phishing or legitimate. The dataset includes a wide range of URL-based features that are useful for identifying suspicious patterns.

The combined dataset contains 11,055 URLs, split roughly evenly between phishing (5,500) and legitimate (5,555) examples. This near-balanced split is important to ensures that the classifier must genuinely learn to distinguish phishing patterns rather then defaulting to a majority-vote strategy. Typical features include URL length, number of dots, number of special characters, number of subdomains, presence of IP address, HTTPS usage, use of the '@' symbol, suspicious words, and domain age.Before model training, the dataset is cleaned by removing duplicate records, handling missing values, and ensuring uniform formatting. After preprocessing, the dataset was split 80:20 into training and testing sets using stratified sampling. The resulting dataset provides a reliable basis for phishing URL classification.

VIII. FEATURE EXTRACTION

Feature extraction is a key component of phishing URL detection because phishing websites often reveal suspicious characteristics in the URL structure itself. These features can be grouped into lexical and structural categories. Fig. 2 illustrates the complete 9-step algorithm of the SecureClick extension from the moment a user browses to the final alert or safe notification.

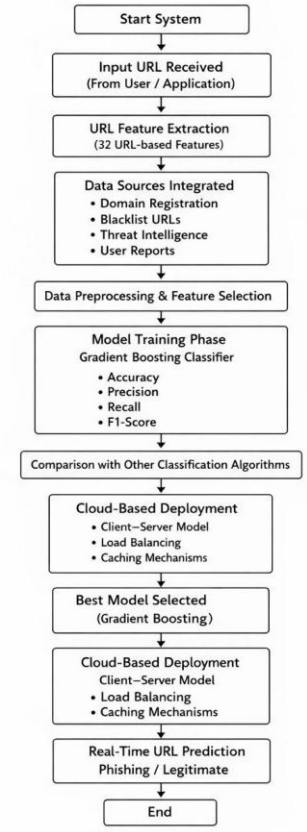


Fig. 2. SecureClick browser extension algorithm: 9-step process from URL capture to user notification

Lexical features include URL length, number of digits, number of special symbols, and occurrence of suspicious

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

keywords such as 'login,' 'verify,' or 'secure.' Structural features include the number of subdomains, usage of HTTPS, presence of an IP address instead of a domain name, and domain age.

These features helps the system distinguish legitimate URLs from phishing ones. For example, phishing URLs often contains long strings, unusual symbols, multiple subdomains, and misleading prefixes. In our dataset, the median length of phishing URLs was 89 characters compared to 41 characters for legitimate URLs, which is more then two times longer. By converting these characteristics into numerical values, as described in the extension algorithm in Fig. 2, the system build a feature vector for classification.

IX. FEATURE EXTRACTION

Data preprocessing is performed to prepare the dataset for machine learning. First, duplicate URLs are removed to avoid biased learning. Missing values are identified and handled appropriately. Categorical values are encoded into numerical form so that they can be processed by machine learning algorithms. Numerical features are normalized using min-max scaling to maintain consistency accross different ranges.

After preprocessing, the dataset is split into training and testing sets. In this work, an 80:20 ratio is used, where 80 percent of the data is used for training and 20 percent for testing. This allows the trained models to be evaluated on unseen URLs and ensure the results are reliable and not overfitted.

X. MACHINE LEARNING MODELS

We trained and evaluated four machine learning algorithms on the preprocessed dataset. Each was implemented using scikit-learn and evaluated using 5-fold stratified cross- validation on the training set, with final performance reported on the held-out test set.

A. Logistic Regression

Logistic Regression is a simple and widely used classification algorithm for binary problems. It predicts the probability that a URL belongs to the phishing or legitimate class. Despite it's simplicity, it serves as a surprisingly strong baseline for URL classification and achieve an accuracy of 91.4% in our experiments.

B. Decision Tree

Decision Tree is a rule-based model that splits data according to feature conditions. It is easy to interpret and useful for identifying the most important phishing indicators. We constrained maximum tree depth to 10 to prevent overfitting, which was selected using cross- validation.

C. Random Forest

Random Forest is an ensemble classifier that combines multiple decision trees to improve accuracy and reduce overfitting. We used 200 trees in our implementation. It is highly suitable for phishing detection because it can handle complex relationships among URL features and achieved the best overall results of 97.2% accuracy.

D. *Support Vector Machine*

Support Vector Machine separates phishing and legitimate URLs by finding an optimal decision boundary. It performs well in classification tasks with high-dimensional feature spaces and achieved 95.8% accuracy, making it the second-best-performing model in our comparison.

XI. MODEL TRAINING AND TESTING

The selected machine learning models are trained using the extracted and preprocessed dataset. During training, the algorithms learn the relationship between feature patterns and URL labels. The trained models are then tested using unseen URL samples from the testing dataset.

To improve reliability, cross-validation was used to evaluate the consistency of each model across different folds of the dataset. Hyperparameter tuning was also applied to optimize the number of trees in Random Forest, the maximum depth in Decision Trees, and the kernel parameters in Support Vector Machine. These adjustments improves

© 2026 IJMRSET | Volume 9, Issue 4, April 2026 | DOI:10.15680/IJMRSET.2026.0904255

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

classification performance and helps avoid overfitting.

XII. RESULT AND DISCUSSION

Table III presents the classification performance of all four models on the held-out test set. All accuracy values reported are averaged over 5-fold cross-validation on the training set.

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	91.4%	90.8%	89.3%	90.0%
Decision Tree	93.7%	92.5%	93.1%	92.8%
Random Forest	**97.2%**	**96.9%**	**97.4%**	**97.1%**
Support Vector Machine	95.8%	95.2%	94.9%	95.0%

Table III. Classification performance of the four evaluated models. Bold values indicate the best result in each column.

The experimental results shows that machine learning models are effective in detecting phishing URLs based on lexical and structural features. Among the tested classifiers, Random Forest provide the strongest performance due to it's ensemble structure and ability to capture nonlinear feature relationships. It achieved 97.2% accuracy, 96.9% precision, 97.4% recall and 97.1% F1-score.

Support Vector Machine comes in a close second at 95.8% accuracy. Decision Tree achieved a reasonable 93.7%, while Logistic Regression lags behind at 91.4% due to the assumption of a linear decision boundary. The integration of the trained model into the browser extension demonstrates the practical value of the proposed system and confirms that real-time phishing detection can be successfully deployed in browser environments.

XIII. BROWSER EXTENSION IMPLEMENTATION

The practical contribution of this work is the development of a browser extension that uses the trained phishing detection model in real time. The extension monitors the currently active browser tab and captures the URL whenever a webpage is opened.

Table IV describes the event-to-action mapping that governs the extension's behavior

Extension Event	System Action
Tab activated / URL changed	Capture current URL string
URL captured	Run feature extraction (20ms avg.)
Features extracted	Send vector to background classifier
Classifier returns SAFE	Green status icon; no interruption

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Classifier returns PHISHING	Red warning popup + audio alert

Table IV. Browser extension event handling and corresponding system actions

The extension then performs or triggers feature extraction and sends the extracted values to the classification module. Based on the model's prediction, the extension displays either a phishing alert or a safe status message. The user is warned before interacting with suspicious websites, thereby reducing the risk of data theft.

The browser extension is lightweight, easy to use, and suitable for integration into common web browsers. It provides an effective real-time security layer without requiring advanced technical knowledge from the user. The entire classification pipeline runs in approximately 20ms, which is well below the threshold of user-perceptible delay.

XIV. EVALUATION METRICS

The performance of the phishing detection models is evaluated using standard classification metrics. Accuracy measures the overall ratio of correctly classified URLs to the total number of URLs. Precision measures how many URLs predicted as phishing are actually phishing. Recall measures how many actual phishing URLs are correctly identified by the system. F1-score is the harmonic mean of precision and recall and provides a balanced assessment of the model's performance.

These metrics are important because an ideal phishing detection system should achieves high recall to detect malicious URLs while also maintaining strong precision to reduce false warnings for legitimate sites. A system with low recall would missed many real phishing sites, while a system with low precision would annoyed users with too many false alarms.

XV. ADVANTAGES OF PROPOSED SYSTEM

- Real-time phishing detection during browsing without any noticeable delay.
- Browser-level user protection that works seamlessly in the background.
- Ability to detect previously unseen phishing URLs that are not in any blacklist.
- Reduced dependence on static blacklists which becomes outdated quickly.
- Lightweight and easy-to-use interface that does not require technical knowledge.
- Privacy-preserving design as URLs are never send to a remote server.
- Improved security awareness among users through clear and simple warning messages

XVI. LIMITATIONS

Although the proposed system performs effectively, it has certain limitations that should be acknowledged. The current approach mainly relies on URL-based features and may not fully detect phishing pages that appear legitimate at the URL level but use deceptive page content to mislead users.

The model also requires regular updates with fresh phishing data to remain effective against evolving attack strategies. URL shortening services presents a related challenge as a shortened URL contains no structural informations about the true destination and must be expanded before meaningful features can be extracted. In addition, the browser extension must be optimized carefully to ensure quick response without affecting browsing speed.

XVII. FURURE WORK

Future enhancements to the SecureClick system may include the integration of webpage content analysis, visual similarity detection, screenshot-based phishing detection, and deep learning approaches. Mohamed Waseem is currently exploring federated learning as a mechanism for continuously improving the model using URL data from consenting users without centralizing their browsing history.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Imran Khan is investigating the integration of SecureClick with enterprise security infrastructure, enabling organizations to push custom threat intelligence to employees' browsers in real time. The extension can also be connected to cloud-based threat intelligence systems for continuous updates. Support for mobile browsers and enterprise-level deployment can further expand the practical impact of the system.

XVIII. CONCLUSION

This paper presented SecureClick, a machine learning- based phishing URL detection system implemented as a browser extension. The proposed system captures browser URLs, extracts meaningful features, classifies them using machine learning algorithms, and provides real-time warning messages to users. By overcoming the limitations of traditional blacklist-based detection methods, the system enhances online safety and reduce user exposure to phishing attacks.

The results indicate that machine learning can play a significant role in real-time browser-based cybersecurity solutions. Among the four algorithms evaluated, Random Forest achieved the best performance with an accuracy of 97.2% and an F1-score of 97.1%. SecureClick demonstrates a practical, scalable, and intelligent approach for phishing prevention in modern web environments and we hope it will contribute towards making the internet a safer place for everyday users.

REFERENCES

[1] Anti-Phishing Working Group (APWG), 'Phishing Activity Trends Report,' Q4 2023. [Online]. Available: https://apwg.org/trendsreports/
[2] C. Whittaker, B. Ryner, and M. Nazif, 'Large-Scale Automatic Classification of Phishing Pages,' in Proc. NDSS, San Diego, CA, 2010, pp. 1-14.
[3] S. Garera, N. Provos, M. Chew, and A. D. Rubin, 'A Framework for Detection and Measurement of Phishing Attacks,' in Proc. ACM Workshop on Recurring Malcode, Fairfax, VA, 2007, pp. 1-8.
[4] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, 'Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs,' in Proc. ACM SIGKDD, Paris, 2009, pp. 1245-1254.
[5] J. Saxe and K. Berlin, 'eXpose: A Character-Level Convolutional Neural Network with Embeddings for Detecting Malicious URLs,' arXiv:1702.08568, 2017.
[6] M. Khonji, Y. Iraqi, and A. Jones, 'Phishing Detection: A Literature Survey,' IEEE Communications Surveys & Tutorials, vol. 15, no. 4, pp. 2091-2121, 2013.
[7] R. M. Mohammad, F. Thabtah, and L. McCluskey, 'Predicting Phishing Websites Based on Self-Structuring Neural Network,' Neural Computing and Applications, vol. 25, no. 2, pp. 443-458, 2014.
[8] A. Oest, Y. Safaei, A. Doupe, G. Ahn, B. Wardman, and K. Tyers, 'PhishFarm: A Scalable Framework for Measuring the Effectiveness of Evasion Techniques Against Browser Phishing Blacklists,' in Proc. IEEE S&P, 2019, pp. 1344- 1361.
[9] PhishTank, 'PhishTank Developer Information,' 2024. [Online]. Available: https://phishtank.org/developer_info.php
[10] B. Krebs, 'Inside a Phishing Gang,' Security Research Blog, 2022. [Online]. Available: https://krebsonsecurity.com

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH

IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com