iJRASET
International Journal For Research in Applied Science and Engineering Technology

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

**Volume:** 12 **Issue:** III **Month of publication:** March 2024

**DOI:** https://doi.org/10.22214/ijraset.2024.59364

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 12 Issue III Mar 2024- Available at www.ijraset.com*

# Safeguarding Cyber Ecosystems: Machine Learning and Data Mining Approaches for Proactive Phishing Detection in Websites

Santosh S. Bagewadi[1], Roshan Arde[2] and Suraj Yadav[3]
*Dr. D. Y. Patil Institute of Technology, Pimpri-411018, Pune*

***Abstract:*** ***The prevalence of cyber threats, notably phishing attacks, presents a significant challenge in the digital realm with the increasing reliance on the internet. Phishing, where victims' credentials are illicitly acquired through deceptive websites resembling legitimate ones, is a particularly concerning form of cybercrime. This paper proposes an innovative system utilizing Machine Learning and Data Mining techniques to detect both established and newly generated phishing URLs without historical behavioral data. The system, embodied in a Chrome browser plugin, operates in real-time to provide immediate protection as users browse web pages. A distinctive feature of this system is its capability to identify phishing websites lacking prior behavioral patterns, ensuring adaptability to evolving cyber threats. Through comprehensive training with an extensive dataset, the model powering the plugin aims for high accuracy in detecting phishing attempts. By deploying advanced Machine Learning methodologies, the system discerns subtle patterns and characteristics intrinsic to phishing URLs, effectively distinguishing them from legitimate websites. The ultimate objective is to enhance cyber security by providing users with a robust and proactive tool against the growing sophistication of phishing attacks, thereby contributing to the ongoing efforts to create a secure online environment amidst the dynamic landscape of cyber threats.***

***Keywords: Cyber threats, Phishing attacks, Phishing Detection, Machine Learning, Data Mining***

## I. INTRODUCTION

Phishing, a deceitful practice, involves the fraudulent acquisition of sensitive information such as passwords and credit card details, often through deceptive emails or messages. These communications, masquerading as legitimate sources, lead users to counterfeit websites designed to mimic authentic ones, where they are prompted to enter personal information [1].Due to the transient nature of phishing websites, traditional detection methods relying on directories of known malicious sites may struggle to keep pace with the emergence of new phishing attempts. Consequently, alternative techniques, such as the random forest classifier, are being explored for more effective detection [2]. Integrating such detection mechanisms directly into web browsers can provide users with real-time warnings when visiting suspicious websites, enhancing their protection against phishing scams [3]. In the transformative year of 2020, the world witnessed an unprecedented reliance on technology fueled by the global pandemic, amplifying the significance of digitalization in people's lives. However, this heightened digital presence also brought forth a surge in cybercrime, notably phishing, a deceptive tactic combining social engineering and technical subterfuge to acquire personal identity data or financial credentials by impersonating trusted websites [4]. In 2020, phishing emerged as the most common cyber-attack, as reported by the FBI. The number of phishing incidents nearly doubled, jumping from 114,702 in 2019 to 241,342 in 2020 [5]. Additionally, the Verizon 2020 Data Breach Investigation Report disclosed that 22% of data breaches in 2020 were attributed to phishing attacks [6].

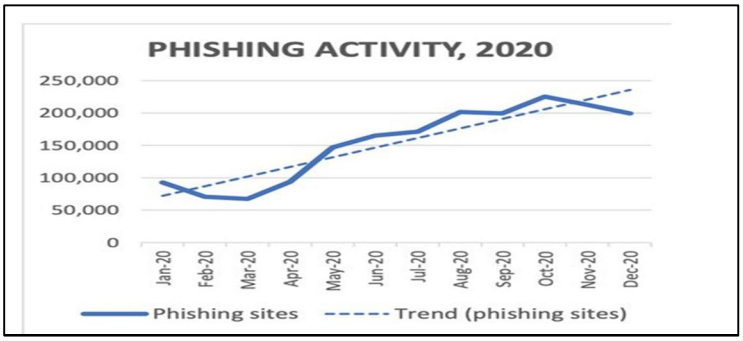



Fig. 1. Phishing Activity – 2020 [4]

*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 12 Issue III Mar 2024- Available at www.ijraset.com*

Reports from 2020 revealed a doubling of security breaches and financial losses attributed to phishing attacks compared to the previous year, with financial institutions being particularly targeted, while attacks against SaaS and Webmail sites fluctuated, and those against E-commerce sites surged. The global pandemic provided fertile ground for cybercriminals to exploit, with phishing attacks capitalizing on the public's focus on COVID-19 through various fraudulent schemes, including fake job offers, fabricated health organization messages, and vaccine-themed phishing attempts. This detailed examination aims to dissect the multifaceted landscape of phishing attacks in 2020, shedding light on their tactics, sector-specific impacts, and the complex interplay between cybercrime and the pandemic to inform effective strategies for fortifying digital defenses in the face of evolving cyber threats [7].

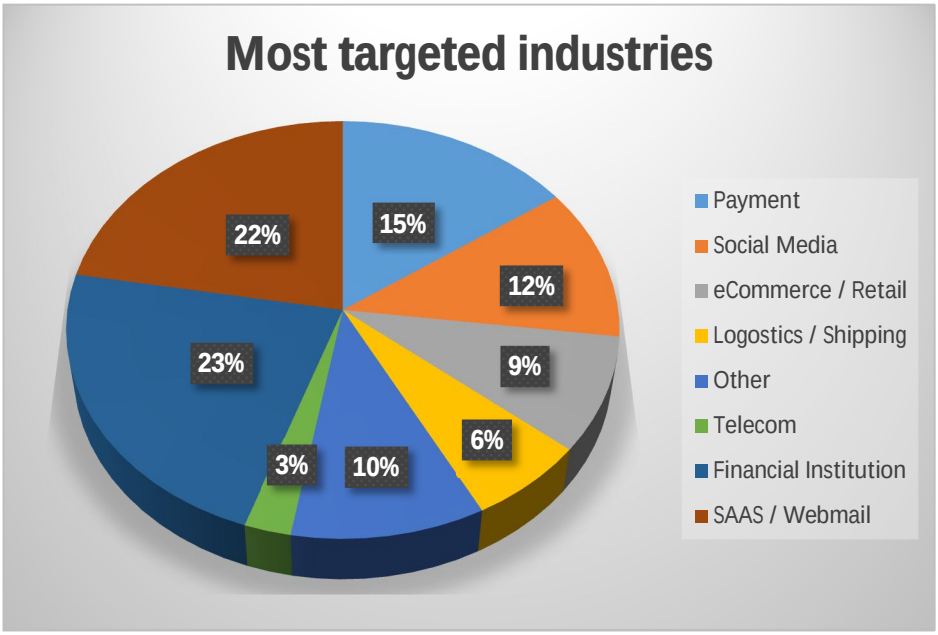



Fig. 2 Most targeted industries, 4Q 2020 [4]

A multitude of studies have delved into the detection and prevention of phishing websites, employing various methodologies and algorithms. Thaker et al. (2018) proposed a method utilizing data mining techniques, specifically the Random Forest Algorithm, to identify phishing websites by extracting features from URLs [1]. Abdullah et al. (2016) emphasized the significance of user education and awareness in combating phishing attacks, advocating for thorough URL verification before accessing websites [3]. Gupta et al. (2018) conducted a comparative analysis of machine learning approaches for phishing detection, highlighting challenges such as accuracy limitations and computational requirements [8]. Varsharani et al. (2016) presented a heuristic algorithm for fast phishing URL detection, aiming to promptly alert users to potential threats [9]. Gillani et al. (2016) developed a hybrid classification model for phishing site detection, intending to enhance accuracy through feature selection mechanisms [10]. Patil et al. (2018) focused on machine learning approaches, with Random Forest achieving notable accuracy rates in detecting phishing websites [11].

Dutta et al. (2021) proposed the use of recurrent neural networks for improved phishing website detection, outperforming traditional methods in identifying fraudulent sites [12]. Additionally, Rao et al. (2023) tested various machine learning methods and found Random Forest to be the most effective in identifying fake websites [13]. Alnemariet et al. (2023) corroborated these findings, identifying Random Forest as the optimal method for detecting phishing domains [14]. Molah et al. (2017) and Korkmaz et al. (2020) also recognized the superiority of Random Forest in phishing website detection, citing its reliability, speed, and accuracy [16,17].

Mahajan et al. (2018) further supported these conclusions, demonstrating Random Forest's superior performance compared to Decision Trees and Support Vector Machines across various dataset splits, emphasizing the importance of ample training data for improved model accuracy [18].

*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 12 Issue III Mar 2024- Available at www.ijraset.com*

## II. MOTIVATION

Phish Tank, a collaborative clearinghouse for phishing data on the internet, offers an open API for integrating anti-phishing data into applications. Similarly, the Google Safe Browsing API follows a directory-based approach and provides an open API akin to Phish Tank. However, relying solely on these approaches proves ineffective against newly developed phishing websites, as directories often fail to remain updated. Additionally, using external services like the Phish Tank API raises privacy concerns due to the transmission of URLs to external servers. To address these limitations, our proposed system not only detects phishing websites but also integrates this detection mechanism directly into a browser plugin. By eliminating the need for external web services, our approach enhances user privacy and ensures that browsing data remains under the user's control. Furthermore, integrating the detection mechanism into a browser plugin enables real-time detection of phishing websites during browsing sessions, offering users immediate protection against evolving cyber threats.

## III. PROPOSED SYSTEM

In our proposed system, we employ a web host model to identify phishing websites. This model relies on a classification algorithm and is trained using a dataset. Once the model is trained, it will be deployed online and will directly interact with a Chrome extension. Detection of phishing websites will be based on both the URL and attributes of the website. This system integrates all functionalities performed on both the client and server sides. On the server side, a classifier model is trained using the random forest algorithm, while on the client side, a Chrome extension is developed and incorporated into the browser. When a user accesses a website using the Chrome browser, the extension retrieves the URL. Subsequently, various attributes of the website are extracted from both the URL and the displayed webpage. These attributes serve as test data for the classifier deployed in the cloud, which has been trained on a dataset of phishing websites. The classifier then determines whether the entered URL is malicious or safe. If it's identified as a phishing website, the user receives an alert warning them that their credentials may be compromised if they proceed. Conversely, if it's deemed a safe website, the user can proceed with their activities on that page without concern.

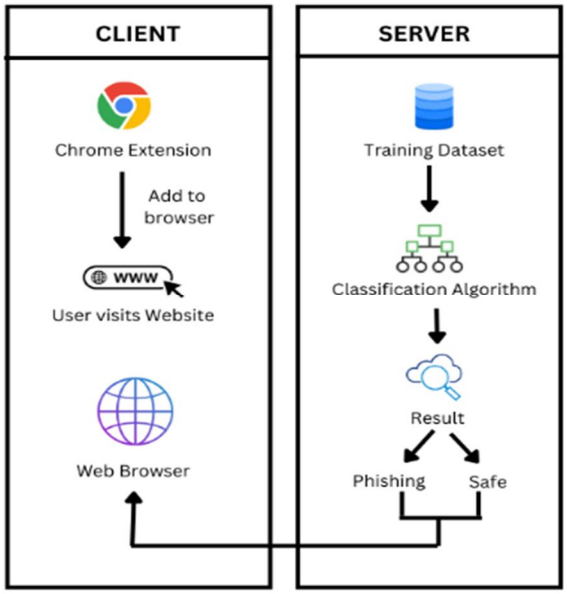



Fig. 3 Architecture design of the system

### A. *Chrome Extension*

Browser extensions are compact software programs designed to enhance the browsing experience by allowing users to customize the functionality and behavior of their web browser, particularly in Chrome. These extensions empower users to personalize their browsing experience according to their specific needs or preferences. They can take various forms, ranging from simple icons, like the Google Mail Checker extension displayed on the right side of the browser, to more complex functionalities that modify or override entire web pages. In our project, we are developing a plugin specifically tailored for Chrome browsers. Once created, this plugin will be seamlessly integrated into the Chrome extension ecosystem, providing users with additional features and capabilities to enhance their browsing experience.

*B. Pre-Processing*

The dataset is obtained from the UCI repository and imported into a numpy array. It comprises 30 features, which need to be condensed for extraction within the browser environment. Each feature is assessed individually within the browser to determine its feasibility for extraction without relying on external web services or third-party tools. Through experimentation, a subset of 17 features is selected from the original 30, ensuring minimal loss in accuracy on the test data. While a larger number of features generally leads to higher accuracy, it also extends the feature extraction time, potentially compromising rapid detection. Thus, a careful balance is struck by opting for a subset of features that maximizes accuracy while minimizing extraction time. Subsequently, the dataset is partitioned into training and testing sets, with 30% allocated for testing purposes. Both the training and testing data are then saved to disk for further analysis and model training.

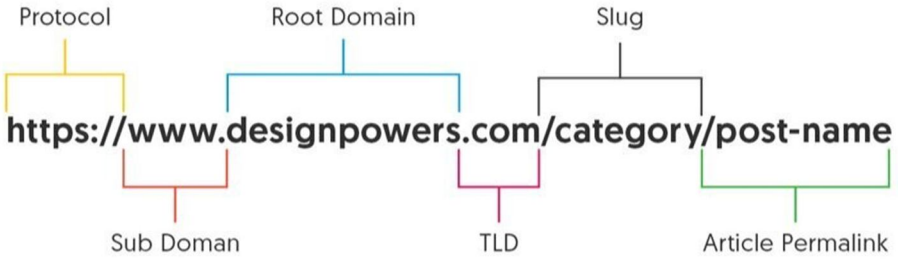



Fig. 4. Structure of URL

| IP Address | Degree of subdomain | Anchor tag href domains |
|---|---|---|
| URL length | HTTPS | Script & link tag domains |
| URL shortener | Favicon domain | Empty server form handler |
| @' in URL | TCP Port | Use of mailto |
| Redirection with '//' | HTTPS in domain name | Use of iFrame |
| -' in domain | Cross domain requests | |

Table.1. Features of URL Extraction

*C. Training Data*

The model we're proposing will be trained using a dataset obtained from the UCI repository. This dataset consists of a total of 11,055 records, with 4,898 identified as phishing websites and 6,157 as legitimate websites. Following preprocessing steps to clean and organize the data, 70% of the dataset will be utilized to train the random forest classifier. This process involves feeding the algorithm with a large portion of the data to enable it to learn and identify patterns effectively

*D. Exporting Module*

During the training phase of machine learning algorithms, they learn their parameter values. In the case of Random Forest, each decision tree acts as an independent learner. This means that each decision tree learns specific values for node thresholds and probabilities for class labels at the leaf nodes. As a result, there's a need to develop a format that can accurately represent the Random Forest in JSON format.

*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 12 Issue III Mar 2024- Available at www.ijraset.com*

## IV. ALGORITHM

For training the proposed system, we primarily rely on the random forest classifier. This classifier plays a central role in processing our training data.

### *A. Random Forest Algorithm*

Random Forests utilize an ensemble learning approach, which means they combine the predictions of multiple individual decision trees to make a final prediction. This technique is akin to a "divide and conquer" strategy, where each decision tree focuses on a different aspect of the data and contributes to the overall prediction. Unlike a single decision tree, where the input starts at the top and follows a single path downwards, Random Forests work by aggregating the predictions of many individual trees. Each tree is trained on a random subset of the data, and the final prediction is determined by combining the predictions of all trees in the forest. This aggregation can be done by averaging the predictions or using a voting mechanism, depending on the type of data being analyzed.

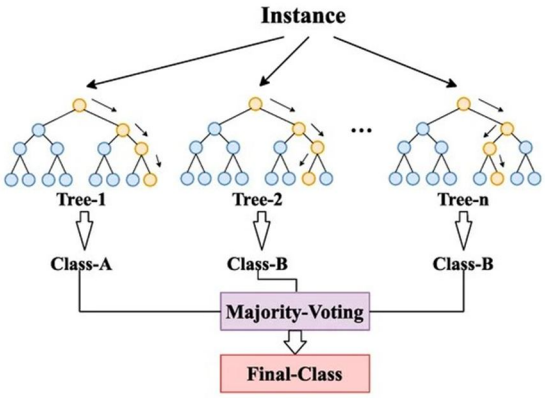



Fig. 5. Random Forest Algorithm

In our system, we employ Random Forest to train a predictive model using a dataset. Specifically, we use a random forest classifier from the scikit-learn library, which consists of an ensemble of 10 decision tree estimators. Each decision tree in the random forest follows the CART (Classification and Regression Trees) algorithm, which aims to minimize the gini impurity a measure of the homogeneity of the nodes in the tree. After training the model on the dataset, we export the trained model to JSON format using an exporting module, making it accessible for future use.

Random Forest is a supervised learning algorithm used for classification and regression tasks. It constructs a "forest" of multiple trees, with each tree contributing to the overall prediction. Generally, a greater number of trees in the forest enhances the robustness of the model and improves prediction accuracy. Therefore, increasing the number of trees in the random forest classifier typically leads to better performance in predicting the likelihood of phishing in a given URL.

Steps Involved:

1) *Step 1:* Begin by randomly selecting samples from the dataset.
2) *Step 2:* Construct a decision tree for each sample and make predictions using these trees.
3) *Step 3:* Perform a vote for each prediction.
4) *Step 4:* Choose the prediction with the most votes as the final result.

### *B. Random Forest Necessity*

1) Works well with various data types and sizes.
2) Has lower variance compared to single decision trees.
3) Highly flexible and accurate in predictions.
4) Maintains accuracy even with missing data.
5) It is robust to overfitting, making it less prone to errors with noisy data.
6) Suitable for both classification and regression problems, making it applicable to various data analysis tasks.
7) Provides insights into the importance of different variables in predicting outcomes.

## V. CALCULATION AND PERFORMANCE

To assess how well a system performs, we use specific measures. These include Accuracy, Precision, Recall, and F1 Score, which are calculated for each machine learning model.

### *A. Precision and Recall*

Precision measures how accurate positive predictions are, while recall gauges the fraction of actual positives correctly identified.
Precision (P) is the ratio of true positives (correctly predicted positives) to the sum of true positives and false positives (incorrectly predicted positives).

The formula to calculate Precision is:

$$(P = (Tp) / (Tp + Fp)) \quad \text{---------- (1)}$$

Recall (R) is the ratio of true positives to the sum of true positives and false negatives (positives missed by the model).
The formula to calculate Recall is:

$$(R = (Tp) / (Tp + Fn)) \quad \text{---------- (2)}$$

### *B. F1 Score*

The F1 Score, or F Measure, finds a balance between precision and recall.
The formula to calculate F1 Score is:

$$(F1 = (2 \text{ times Precision Recall}) / (\text{Precision} + \text{Recall})) \quad \text{---------- (3)}$$

It's like a blended measure, considering both precision and recall in a single score.

### *C. Accuracy*

Accuracy is basically how often a classification model, like Random Forest, gets its predictions right out of all the predictions it makes.
So, if you have 100 predictions and 80 of them are correct, the accuracy would be 80%.

The formula to calculate accuracy is:

$$(\text{Accuracy}) = (\text{Number of Correct Predictions}) / (\text{Total Number of Predictions}) \quad \text{---------- (4)}$$

## VI. IMPLEMENTATION AND RESULT

We used a tool called Scikit-learn to bring in machine learning algorithms. Then, we split the dataset into two parts: a training set and a testing set. We tried three different ratios: 50:50, 70:30, and 90:10, meaning we used half, 70%, or 90% of the data for training and the rest for testing.
Next, we trained each classifier using the training set and tested their performance using the testing set. We measured their performance using three metrics: accuracy score, false negative rate, and false positive rate. These metrics help us understand how well the classifiers are doing in making correct predictions and avoiding mistakes.

| Dataset Split ratio | Classifiers | Accuracy Score | False Negative Rate | False Positive Rate |
|---|---|---|---|---|
| 50:50 | Decision Tree | 96.61 | 3.59 | 2.83 |
| | Random Forest | 96.62 | 3.59 | 2.81 |
| | Support vector machine | 96.30 | 5.16 | 1.98 |

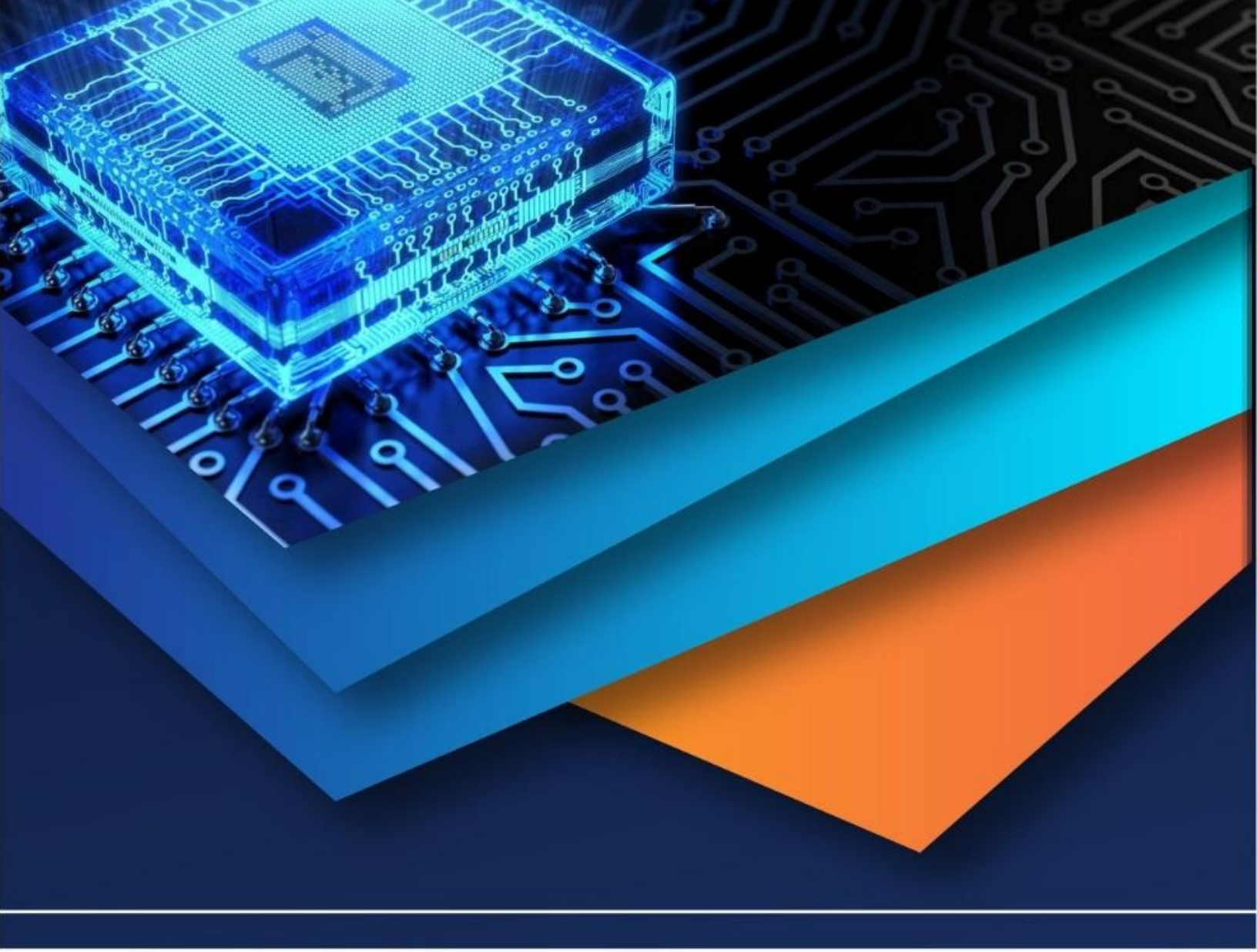
| | | | | |
|---|---|---|---|---|
| 70:30 | Decision Tree | 96.70 | 3.33 | 2.89 |
| | Random Forest | 96.74 | 3.25 | 2.88 |
| | Support vector machine | 96.30 | 5.03 | 2.07 |
| 90:10 | Decision Tree | 97.01 | 3.08 | 2.56 |
| | Random Forest | 97.04 | 3.04 | 2.51 |
| | Support vector machine | 96.41 | 4.63 | 2.24 |

## VII. CONCLUSION

Phishing Detection is an important aspect of keeping online activities safe and secure. This paper explore different methods, like using machine learning and analyzing website URLs, to spot potential scams. By carefully choosing what data to look at and using smart techniques to simplify things, research studies make these detection process faster and more accurate. It also provides useful information for others starting similar research, helping them understand how to detect phishing better. Overall, the goal is to improve cyber security by giving users the tools to recognize and avoid online threats, while still respecting their freedom to make choices online. Overall, it promotes a balanced approach to cyber security by combining proactive warning systems with user autonomy.

## REFERENCES

[1] Mehek Thakera, Mihir Parikhb, Preetika Shettyc, Vinit Neogid, Shree Jaswale ”Detecting the phishing website using data mining”.2nd International conference on Electronics, Communication and Aerospace Technology (ICECA 2018) ,Pp.1876-1879
[2] J. Crowe, ‘Phishing by the Numbers: Must-Know Phishing Statistics 2016’, 2016. [Online]. Available: https://blog.barkly.com/phishingstatistics-2016
[3] Ahmed, Abdulghani Ali, and Nurul Amirah Abdullah. ”Real time detection of phishing websites.” Information Technology, Electronics and Mobile Communication Conference (IEMCON), 7th Annual. IEEE, 2016, Pp.1-6
[4] Anti-phishing Working Group (APWG) Phishing Activity Trends Report 4th quarter 2020, https://docs.apwg.org/reports/apwg trends report q4 2020.pdf
[5] FBI Internet Crime Report 2020, https://www.ic3.gov/Media/PDF/AnnualReport/2020 IC3Report.pdf
[6] Verizon 2020 Data Breach Investigation Report,https://enterprise.verizon.com/resources/reports/2020-databreachinvestigations- report.pdf
[7] World Health Organization, Communicating for Health, Cyber Security, https://www.who.int/about/communications/cyber-security
[8] Jain, Ankit Kumar, and B. B. Gupta. ”Comparative analysis of features based machine learning approaches for phishing detection.” Computing for Sustainable Global Development (INDIA.Com), 2016 3rd International Conference on.IEEE,2016, Pp.2125-2120
[9] Hawanna, Varsharani Ramdas, V. Y. Kulkarni, and R. A. Rane.”A novel algorithm to detect phishing URLs.” Automatic Control and Dynamic Optimization Techniques (ICACDOT), International Con ference on. IEEE, 2016 pp.548-552
[10] Tahir, M. Amaad Ul Haq, et al.”A Hybrid Model to Detect PhishingSites Using Supervised Learning Algorithms.” Computa- tional Science and Computational Intelligence (CSCI), 2016 Inter- national Conference on. IEEE, 2016.Pp.1126-1133
[11] Vaibhav Patil et al (2018) “Detection and Prevention of Phishing Websites using Machine Learning Approach” 978-1-5386-5257-2/18/$31.00 ©2018 IEEE
[12] Ashit Kumar Dutta et al (2021) “Detecting phishing websites using machine learning technique” https://doi.org/10.1371/journal.pone.0258361
[13] Mr. P. P Nagaraja Rao et al (2023) “Phishing Website Detection using Machine Learning” International Journal of Advanced Research in Science, Communication and Technology DOI: 10.48175/IJARSCT-8318
[14] Shouq Alnemariet al (2023) “Detecting Phishing Domains Using Machine Learning”, Appl. Sci. 2023, 13, 4649. https://doi.org/10.3390/app13084649
[15] William Koehrsen, ‘Random Forest Simple Explanation’, 2017. https://medium.com/@williamkoehrsen/randomforest-simple-explanation-377895a60d2d
[16] Abdulhamit Subasi, Esraa Molah, Fatin Almkallawi, Touseef J. Chaudhery, “Intelligent phishing website detection using Random Forest classifier,” International Conference on Electrical and Computing Technologies and Applications(ICECTA), 2017
[17] Mehmet Korkmaz, Ozgur Koray Sahingoz, Banu Diri, “Detection of phishing websites by using machine learning-based URL analysis,” 11nth International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2020
[18] Rishikesh Mahajan, and Irfan Siddavatam, “Phishing website detection using machine learning algorithms,” International Journal of Computer Applications(0975-8887), vol. 181, no. 23, 2018



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY