International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 05 | May - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

ENHANCING SECURITY: AN INTELLIGENT PHISH CATCHER WITH SAFE SEARCH ON BROWSERS

Dr.T.MAHESH SELVI M.E.,P.hd.,

RAJESWARI S[2], SHALINI S[3], DHARANIKA S[4]

Department of Computer Science and Engineering University College of Engineering, Thirukkuvalai

(A constituent College Of Anna University::Chennai and Approved by AICTE, New Delhi)

ABTRACT

A frequent and major hazard to cyber security is a malicious URL or website. Of course, search engines end up becoming the foundation of information management. However, our users are under great jeopardy due to the proliferation of fraudulent websites on search engines. Most existing malware detection systems concentrate on particular types of attacks. Blacklist-based browser add-ons that are now available are helpless against a vast number of websites. As a result, any data that leaves the client side must be adequately masked such that the server is unable to deduce any useful information from the masked data. The first PPSB service is proposed here. Strong security guarantees are offered, which are absent from current SB services. Specifically, it takes over the ability to identify risky URLs while simultaneously safeguarding the user's privacy (search history) and the proprietary assets of the blacklist provider (the harmful URL list). This study suggests a paradigm that encrypts users' private information to protect it from outside analysts and service providers. Additionally, it fully supports the selective aggregate functions needed for the investigation of online user behaviour while maintaining differentiated privacy. Here, user search activity data is encrypted using the AES encryption technique. Only authorized users who have completed the verification process and shared a secret key can decrypt the history. The search engine administrator can be informed of any fraudulent URL by using the user feedback mechanism. An administrator may regularly update the database of harmful URLs.

Keywords: Intelligent Phish Catcher, SVM algorithms and Python**.**

1. INTRODUCTION

Phishing mimics the traits and appearance of emails, making it appear identical to the real thing. It looks like the one from the reliable source. The user believes that this email is authentic and originates from a reputable company or organization. This compels the user to use the links provided in the phishing email to visit the phishing website. These phishing websites are designed to mimic the design of legitimate company websites. Phishers compel victims to provide personal information by sending scary messages, confirming account messages, and other tactics. This allows the victims to fill out the necessary fields, which they can then misuse.

They manipulate the circumstances such that the user is forced to visit their spoof website. We should employ labelled data with samples from both genuine and phishing areas during the training phase. This way, identifying the phishing domain won't be hampered by classification. Using a data set during the training phase is essential to creating a functional detection model.

The samples we mark as phishing should only be identified as phishing as we should only utilize samples whose classes we are familiar with. In a similar vein, samples that have been certified as authentic will be recognized as authentic URLs. These characteristics must truly be present in the dataset that will be used for machine learning. Numerous machine learning algorithms exist, and as we have previously shown in the previous chapter, each algorithm has a unique operating mechanism. The current approach forecasts the accuracy of phishing URL identification by utilizing any appropriate machine learning technique.

Although the current method is accurate, it is still not the greatest because phishing attacks are a serious problem that need to be solved. It is not a smart idea to employ a single machine learning algorithm to increase prediction accuracy in the current system, as it only uses one method to forecast accuracy. A malicious SB service provider is interested in finding out if a user is viewing a specific website, such as one that contains political news. Sending all of the visited URLs—in encrypted, hash value, or plaintext format—to a remote server via the web browser is one method to accomplish this. But this behavior can be seen by keeping an eye on and examining the browser, for example, by applying the taint analysis technique.

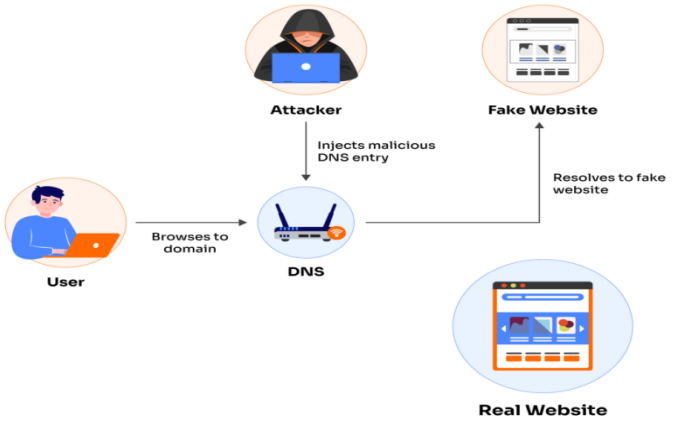


Fig 1.1 Phishing Detection System

A malevolent SB service provider aims to ascertain whether a user is browsing a specific website, like one that holds political data. The web browser can accomplish this, for example, by sending all visited URLs—in plaintext, hash value, or encrypted format—to a remote server. Nonetheless, tracing and examining the browser—for example, by using the taint evaluation technique—may reveal this activity. Afterwards, after a user views the website (or similar URLs that comprise a few decompositions),

The remote SB server may get the matched hash prefixes. The URL (or region) that the user navigated can be inferred by the server using the previous data of the prefix filter (i.e., the mappings between the hash prefixes and their respective URLs). It provides robust security guarantees that current SB services do not offer. More precisely, it inherits the ability to recognize dangerous URLs.

The SB carrier issuer can insert the 32-bit hash prefixes of all of its decompositions, such c01e362f, to track a certain URL. It can then provide its clients the most recent version of this prefix filter. Specifically, to monitor a given URL, the SB carrier issuer can append the 32-bit hash prefixes of all of its decompositions, such as c01e362f. Then, it can give the clients the most recent iteration of this prefix filter. Simultaneously safeguards the user's privacy (browsing history) and the private assets of the blacklist provider (the list of dubious URLs). There are a few other shortcomings to this strategy, like the incapacity to create URL metadata when the server receives several prefixes for a particular URL.
To solve this capacity issue, PPSB provides users with an adaptable method for adding or removing blacklist providers. The phony URL and phrase should be added to this blacklist stored by the admin. The user is additionally permitted to recommend dangerous websites regarding blacklist information. The malware detection machine on this device uses a supervised machine learning technique to find malware. The idea of a signature-based detection system is expanded upon by the SVM classification with malware detection system, which uses an aggregate of conduct tracking technique. It uses the executable run-time traces to evaluate malware both statically and dynamically. The ability to compare picture functions based entirely on the malicious website and the genuine website is another feature of image-based harmful detection. This version also has seek records security, which encrypts crucial data to shield users' private information from prying eyes and the aggregate provider. Additionally, it fully supports the selective combination functions needed for the analysis of online consumer behaviour and the protection of individual privacy.

2. WEB SECURITY

The Internet poses a significant risk! We frequently read about websites that are taken down by denial of service attacks or that have had their homepages altered to include content that is harmful. Millions of passwords, email addresses, and credit card numbers have been exposed in other high-profile cases, putting website users at danger for financial loss as well as personal shame. Ensuring the security of your website takes careful design work in all areas, including your web application, web server settings, password creation and renewal rules, and client-side code. The good news is that if you're using a server-side web framework, it will almost certainly enable "by default" strong and well-thought-out security mechanisms against some of the more prevalent attacks, even though all of that sounds very dire. You can reduce the impact of other assaults by configuring your web server, such as by turning on HTTPS. Lastly, you can use publicly accessible vulnerability scanners to determine whether you have committed any glaring errors.

2.1Common attacks/vulnerabilities

Click jacking:

In this assault, a malevolent user directs clicks intended for a visible top-level site to a hidden one below. For example, this technique may be used to show a genuine bank website while capturing the login credentials and storing them in an invisible under the attacker's control. Another way to trick a user into clicking a button on a website they can see is by using click jacking, which can cause them to accidentally click on an entirely different button. By adjusting the relevant HTTP headers, your website can defend against being included as an I frame on another website.

Denial of Service (DOS):

DOS attacks are often carried out by bombarding a target website with fictitious requests, which prevents authorized users from accessing the website. The requests could just be too many, or they could each use a lot of resources (such sluggish readings or the downloading of big files)...Typically, Dos defences function by detecting and obstructing "bad" traffic while permitting the delivery of valid messages. Usually found either before or inside the web server, these defences are not a component of the online application itself.

Directory Traversal (File and disclosure):

A malevolent user tries to gain access to areas of the web server file system that they shouldn't be able to in this attack. When a user is able to pass filenames that contain file system navigation characters (like.../../), this vulnerability arises. Sanitizing input before using it is the solution.

File Inclusion:

A user can designate a "unintended" file for execution or display in data sent to the server using this attack. This file may be loaded and then run client-side or on the web server (perhaps resulting in an XSS attack). Sanitizing input before using it is the solution.

Command Injection:

Through command injection attacks, a malevolent person can take control of the host operating system and issue arbitrary system commands. Cleaning up user input before it's used in system calls is the answer.

AIM AND OBJECTIVE

The following is a list of the goals for the suggested unsafe (or malicious) URL detection and search history encryption method.

• The goal of this project is to examine the website's URL in order to look for any suspicious patterns, including misspelled domain names or odd characters.

• It is possible to create a system that can identify phishing websites in search results and alert people to avoid them.

• Supporting specific aggregate functions is necessary to provide differential privacy and analyse online user behaviour.

• Using a combination of behaviour monitoring techniques, the SVM-based malware detection system expands on the concept of signature-based detection systems.

• These algorithms can identify novel and developing phishing techniques since they can be taught with previous data.

• The secure transmission of the URL of the searched material to the server. The URL has been encrypted using the AES encryption method.

3. RELATED WORK:

[1] Title: PHIDMA: A MULTI-FILTER APPROACH PHISHING DETECTION MODEL AUTHORS: K. S. KUPPUSAMY, GUNIKHAN, AND SONOWAL. Provide the PhiDMA (Phishing Detection using Multi-filter Approach) multilayer model for phishing detection. The auto upgrade whitelist layer, URL characteristics layer, lexical signature layer, string matching layer, and accessibility score comparison layer are the five layers that make up the PhiDMA model. A working prototype of the proposed PhiDMA model has an accessible interface to allow those with vision impairments to use it without any difficulty. One of the biggest obstacles to using anti-phishing solutions for those with vision impairments is their accessibility. This paper's primary goal is to provide an anti-phishing model for people with vision impairments. The suggested model makes use of the website page's content and URL properties. The suggested model includes two Boolean outcomes: passed and failed. The model's overall structure is multilayer, with labels A, B, C, D, and E assigned to each layer. The result indicates that the URL is not recognized as phishing.

The model's layers function as a pipeline, with each layer passing before the subsequent layer has the chance to confirm. This interface was created with people with visual impairments in mind using a variety of standards. The most important requirements are that people who are blind or visually impaired may prefer less system contact. The shortcut keyboard is offered for interacting with the model, and it only allows for one interaction with the user. The model gets the input and runs it through all of the filters when the keyboard shortcut is pressed. Lastly, audio is used to indicate the outcome. The model uses an audio-based indicator. In our previous study, we analysed anti-phishing browser add-ons and found that the majority of them used a colour-based clue to provide the phishing alarm (green for real and red for phishing). Users that rely solely on the system's auditory cues are completely cut off from this colour-based cue.

[2] Title: TRANSFORMERS' BIDIRECTIONAL ENCODER REPRESENTATIONS: A PHYSICAL ANALYSIS Authors of ODYSSEY: MANIT MISHRA, SHIVAJI, AND ALAPARTHIDESCRIPTION: Using Sent Word Net as the basis for the unsupervised lexicon-based model, logistic regression as the traditional supervised machine learning model, long short-term memory (LSTM) as the supervised deep learning model, and bidirectional encoder representations from transformers (BERT) as the advanced supervised deep learning model, the study aims to compare the relative efficacy of these four sentiment analysis techniques. We employ the Sent Word Net lexicon, logistic regression, and publically accessible labelled corpora of 50,000 movie reviews that were first published on the Internet Movie Database (IMDB) for our analysis.

BERT and LSTM. While BERT was operated on a GPU-based system, the first three models were executed on a CPU-based system. Accuracy, precision, recall, and F1 score were used to assess the sentiment categorization performance. The paper presents two important findings: (2) The indisputable dominance of the pre-trained advanced supervised deep learning BERT model in sentiment analysis from text data; (3) The relative effectiveness of four extremely sophisticated and popular sentiment analysis methodologies. This study offers important insights into the comparative classification performance evaluation of important sentiment analysis approaches, such as the recently established BERT, to academics and professionals in the analytics industry working on text analysis.

This is the first study to compare various sentiment analysis models such as Sent Word Net, LSTM, and logistic regression with the sophisticated pre-trained supervised deep learning model of BERT. Sent Word Net was used for the unsupervised lexicon-based model, logistic regression was used for the supervised machine learning model, LSTM was used for the supervised deep learning model, and pre-trained BERT was used for the advanced supervised deep learning model to classify sentiment. The pre-trained BERT model is unquestionably the best of the four models when it comes to sentiment classification.

[3] Title: Recommending Algorithm for Collaborative Filtering Based on TF-IDF and User Characteristics, JIANJUN, YU CAI, GUANGYI TANG, AND YINGJUAN XIE are the authors. DESCRIPTION: Based on user characteristics and the Term Frequency-Inverse Document Frequency (TF-IDF) technique, a better collaborative filtering algorithm is developed. The user similarity in the proposed technique is initially computed using an improved TF-IDF method based on rating data. Second, a fuzzy membership method is utilized to determine the user similarity based on the multi-dimensional attributes information of the users.

Then, using an adaptive weighted technique, the two user similarities mentioned above are fused. Both the rating data and the user attributes are taken into account in full in the suggested strategy. This paper's contribution can be summed up as follows: (1) The user similarity matrix is computed using the TF-IDF method based on the rating data in order to penalize the influence of popular things on user similarity and enhance the

mining capacity of unpopular items. The suggested approach, which combines several user dimension characteristics data to compute user similarity based on a fuzzy membership function, takes into account all user attributes in order to address the cold start issue. (3) To create a new user comprehensive similarity for recommendation method, an adaptive weighted algorithm is proposed that fuses the two types of user similarities found in the previous two steps. Finally, tests are run using actual data sets to assess how accurate the suggested recommendation model is. In this research, the TF-IDF statistical method is applied to solve the impact of popular items on user similarity. This method optimizes the formula to fit the recommendation model. Simultaneously, a better approach for calculating the similarity of user characteristics is given, which utilizes the user characteristics data and resolves the cold start issue. Lastly, using Movie lens data sets, this paper does offline tests.

[4] Title: Using Machine Learning to Detect Phishing URLs; Authors: Ahammad, SK Haseee, Sunil D. Kale, and Mr. Dipumar BAHADUR JANG

DESCRIPTION:Provide a method for identifying these websites using machine learning techniques that are cantered on the characteristics and actions of the recommended URL. To identify harmful websites, the online security community has developed blacklisting services.

These blacklists are produced using a range of techniques, including heuristics for site inspection and manual reporting. Many harmful websites unintentionally avoid blacklisting because of their recentness, lack of evaluation, or inaccurate evaluation. Algorithms like Support Vector Machine (SVM), Random Forests, Decision Trees, Light GBM, Logistic Regression, and Logistic Regression are used to build a machine learning model that determines whether a URL is malicious or not. A cybercriminal will design a website that mimics the genuine article, and all of the content will match the absolute URL.

On other websites, the URL will show up as an advertising. When the user inputs their credentials, fraud will occur. Another method involves sending the user a malicious URL via email. When the user attempts to open the URL, a dangerous virus is downloaded, giving hackers access to the data they need to carry out their crimes. We must extract certain properties from malicious and benign URLs in order to differentiate between them. A portion of a harmful URL's features must be extracted, and these elements must then be compared to ascertain whether the URL is malicious or benign. As a result, all of the previously covered techniques may be used to develop a machine learning model. For testing and training, the model and 80% of the dataset were used for training, and the remaining 20% for testing. The machine learning techniques are Random Forest, Decision Tree, Logistic Regression, Light GBM, and SVM. used to analyze to determine whether such a URL is fraudulent or not.

[5] TITLE: HYBRID RULE-BASED SOLUTION FOR PHISHING URL DETECTION USING CONVOLUTIONAL NEURAL NETWORK.GHADAH ALDABBAGH, ABDULLAH ALGHAMDI, DANIYAL ALGHAZZAWI, MOURTAJI, YOUNESS, AND MOHAMMED BOUHORMA are the authors .DESCRIPTION: Using 37 features culled from six distinct approaches—the black listed technique, lexical and host method, content method, identity method, identity similarity method, visual similarity method, and behavioural method—deploy a phishing detection mode. Additionally, a comparative study was conducted between various deep learning and machine learning models, such as CNN and MLP (multilayer perceptron's) and machine learning models like CART (decision trees), SVM (support vector machines), or KNN (K-nearest neighbours). Our technique is innovative in that it is based on rational criteria to enhance the logic of treatments because it is possible to detect a URL in some situations without going through the entire process. But given the method's applicability, accuracy, and dependability, the study recommended going through the entire procedure to update the model and gather fresh data regarding how the phishing phenomena evolves.

Using dynamic features extraction is made easier by it. Deep learning networks have demonstrated a remarkable capacity to address the issue of data-driven training, particularly when dealing with large datasets. Typically, computer vision applications use it to solve picture identification and classification issues. On the other hand, research in the field of machine learning is predicated on sound conceptual methods. Deep learning models, particularly CNN, have not only shown very good performance and accuracy in the binary classification for phishing URL detection, but have also assisted in the reduction of error rate. However, trainings associated with deep learning networks require more resources due to the massive mathematical calculations that can be made.

4. EXISTING SYSTEM

A client-side defines mechanism based on machine learning techniques has been built into the current system to recognize fake websites and shield consumers from phishing assaults. Our machine learning approach is implemented as a proof of concept in a Google Chrome extension called Phish Catcher, which categorizes URLs as trustworthy or suspicious. The random forest classifier determines whether or not a login online page is fake when the algorithm receives four distinct kinds of web properties as input.

Conventional classifiers employed methods such as host-based and lexical analysis of URLs, whitelisting and blacklisting, and online learning methodologies. Because blacklisting does not predict the state of previously visited URLs, it is ineffective on its own. Additionally, lexical-based models and whitelisting produced more accurate classifiers than online strategy-based ones .Following the extraction of web page features, a random forest classifier model is chosen based on performance measures including efficiency, accuracy, and latency. After that, the supervised machine learning method was used to train the classifier.

The chosen model was then given the retrieved features to finish the learning process. The model is prepared for testing once the learning process is finished.

This methodology offers to conduct the categorization in the browser itself, in contrast to the usual methods. By resolving latency problems and enhancing tool efficiency, it closes the gaps in the current online applications. Our plug-in's user interface has been simplified to aid in user comprehension. A phishing alert appears on the screen and a drop-down menu highlighting the relevant phishing aspects of the entered URL appears when a user enters a phished URL. Thirty features make up the feature-set, which is divided into four groups, each of which is recognized as a decision tree. The combined results of the decision trees are used by the random forest classifier to distinguish between legitimate and fraudulent login pages.

4.1 DISADVANTAGES

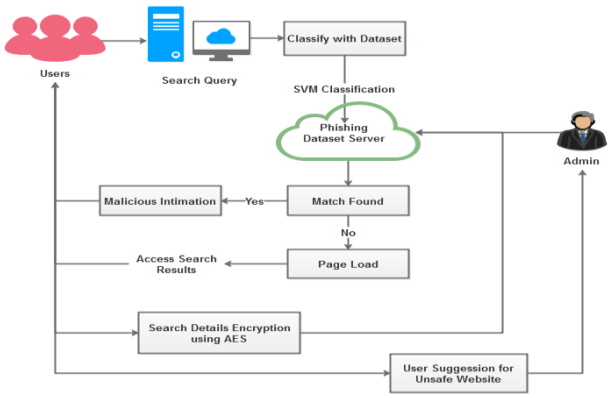
• Erroneous classifications of examples, or noise in the training labels, might adversely affect the accuracy of the model.
• An unbalanced dataset may make it harder to identify the minority class with accuracy.
• Privacy issues may arise when data is gathered and processed on client devices for machine learning.
• Users may become irate and lose trust if websites are mistakenly blocked from access.

5. PROPOSED SYSTEM

By adding several phony or secure URLs or lengthening the server-side delay, a malevolent entity could exploit PPSB to deteriorate the client-side user experience. PPSB offers a flexible option for customers to add or remove blacklist providers in order to handle this potential problem. The phony URL and keyword could be added by the admin to this blacklist store. Users are also permitted to propose dangerous websites for the black list. This system's malware detection method finds malware through supervised machine learning. With a combination of behaviour monitoring techniques, the SVM based malware detection system expands on the concept of signature based detection systems. It uses the executable run-time traces to analyse malware both statically and dynamically. Additionally, this strategy offers search data security, which protects users' private information from outside analysts and the aggregation service provider by encrypting it. Additionally, it fully supports the selective aggregate functions needed for the investigation of online user behaviour while maintaining differentiated privacy. Additionally, the privacy of the user's search history is provided by the proposed solution. to offer security through the use of the AES encryption technique.

5.1ADVANTAGES

• Strong security is ensured by the use of differential privacy and AES encryption in this paradigm.
• Neither the server nor a malevolent user can forecast how users will utilize websites while they are online.
• Stop users from visiting dangerous websites.



BLOCK DIAGRAM

6. IMPLEMENTATION

1. Anti-Phishing Module: To find possible phishing sites, use machine learning algorithms to examine the content, architecture, and past performance of websites. Use heuristics, such as misspelled domain names or variants of popular websites, to identify suspicious URLs.

2. Safe Search Integration : Include a secure search function that removes offensive or dangerous content from search engine results. Work along with trustworthy search engines to make sure that material and websites are correctly categorized.

3. User Feedback Mechanism : Provide a way for users to report dubious websites or erroneous positives or negatives so that the system can be improved over time.

4. Browser Compatibility: Assure compatibility with widely used platforms and browsers to give users a smooth and consistent experience.

5. Real-Time Analysis : Use real-time web page analysis to spot any changes or irregularities that might point to possible dangers.

7. CONCLUSION

Use machine learning techniques to develop a malicious URL detection mechanism in the suggested study. This focuses on using encrypted blacklist storage to identify dangerous website URLs and phrases. A few well chosen characteristics, it is said, can be utilized to distinguish between fraudulent and trustworthy websites. Numerous aspects have been chosen, including URLs and keywords. In the suggested work, a service provider with a superior blacklist—which might be updated more regularly or just have a larger number of items—is involved. Users may also be able to get these datasets by directly sharing blacklists with servers in an unpredictable manner. Accurately identify fraudulent websites with the aid of an effective classification technique, then block people from visiting them. Additionally, this offers a secure encryption method to prevent unauthorized access to search history. The search results that have been kept in the database are protected.

8.REFERENCE

[1] Sonowal, Gunikhan, and K. S. Kuppusamy. "PhiDMA–A phishing detection model with multi-filter approach." Journal of King Saud University-Computer and Information Sciences 32, no. 1 (2020): 99-112.

[2] Alaparthi, Shivaji, and Manit Mishra. "Bidirectional Encoder Representations from Transformers (BERT): A sentiment analysis odyssey." arXiv preprint arXiv:2007.01127 (2020).

[3] Ni, Jianjun, Yu Cai, Guangyi Tang, and Yingjuan Xie. "Collaborative filtering recommendation algorithm based on TF-IDF and user characteristics." Applied Sciences 11, no. 20 (2021): 9554.

[4] Ahammad, SK Hasane, Sunil D. Kale, Gopal D. Upadhye, Sandeep Dwarkanath Pande, E. Venkatesh Babu, Amol V. Dhumane, and Mr Dilip Kumar Jang Bahadur. "Phishing URL detection using machine learning methods." Advances in Engineering Software 173 (2022): 103288.

[5]Mourtaji, Youness, Mohammed Bouhorma, Daniyal Alghazzawi,Ghadah Aldabbagh, and Abdullah Alghamdi. "Hybrid rule-based solution for phishing URL detection using convolutional neural network." Wireless Communications and Mobile Computing 2021 (2021): 1-24.

[6] Odeh, Ammar, Ismail Keshta, And Eman Abdelfattah. "Machine Learning techniques for Detection Of Website Phishing: A Review For Promises And Challenges." In 2021 Ieee 11th Annual Computing And Communication Workshop And Conference (Ccwc), Pp. 0813-0818. Ieee, 2021.

[7] Butt, Muhammad Hassaan Farooq, Jian Ping Li, Tehreem Saboor, Muhammad Arslan, And Muhammad Adnan Farooq Butt. "Intelligent Phishing URL Detection: A Solution Based On Deep Learning Framework." In 2021 18th International Computer Conference On Wavelet Active Media Technology And Information Processing (Iccwamtip), Pp. 434-439. Ieee, 2021.

[8] Tang, Lizhen, And Qusay H. Mahmoud. "A Survey of Machine Learning-Based Solutions for Phishing Website Detection." Machine Learning and Knowledge Extraction 3, No. 3 (2021): 672-694

.[9] Purbay, Madhurendra, And Divya Kumar. "Split Behavior of Supervised Machine Learning Algorithms for Phishing URL Detection." In Advances In Vlsi, Communication, And Signal Processing: Select Proceedings Of Vcas 2019, Pp. 497-505. Springer Singapore, 2021.

[10] Wazirali, Raniyah, Rami Ahmad, And Ashraf Abdel-Karim Abu-Ein. "Sustaining Accurate Detection of Phishing URLs Using SDN and Feature Selection Approaches." Computer Networks 201 (2021): 108591