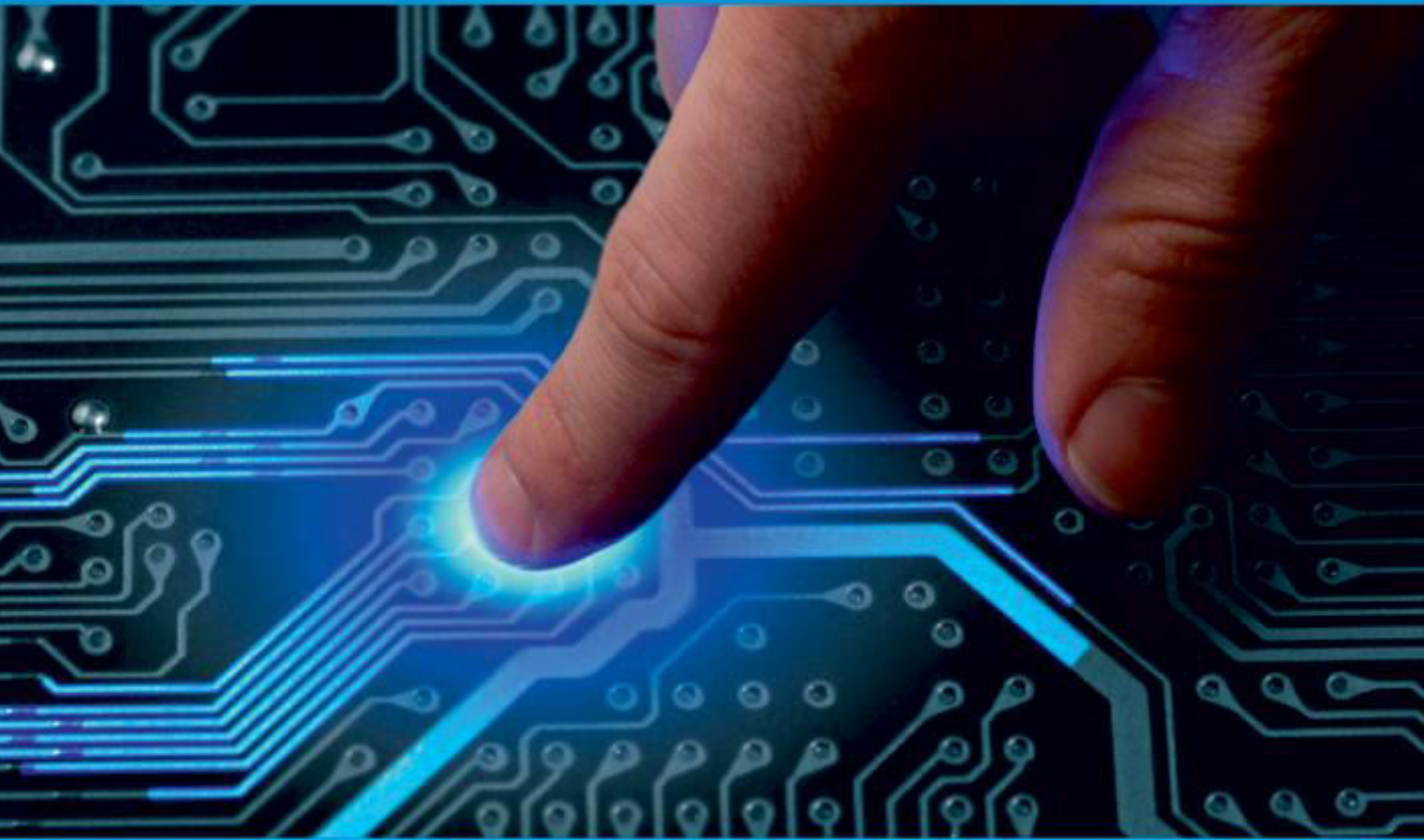
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 7, July 2021

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.542

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907226 |

Machine Learning Approach for Phishing Websites Detection

Prof. T Auntin Jose, Sushma M, Navya J S, Lynette Francis Mathew, Meghana L

Assistant Professor, Department of Computer Science and Engineering, Rajarajeswari College of Engineering, Kumbalgodu, Bangalore, India

Department of Computer Science and Engineering, Rajarajeswari College of Engineering, Kumbalgodu, Bangalore, India

ABSTRACT: Phishing assaults, which are focused on social engineering and malware, are a type of cybercrime that is prevalent in today's society. It is one of the most dangerous risks that every person and organization must deal with. Users find information on the internet using URLs, which are also called web links. The review raises phishing assault awareness, detection, and encourages phishing practice. Phishers use email or messages as a weapon to deceive people by sending URL links to them. Companies and individuals are unable to detect all phishing emails or messages due to the large number of them received each day. Various reviews are presented here for detecting phishing attacks using machine learning. It is used to detect phishing or malicious online URLs. Using the naive Bayes to predict.

KEYWORDS: Machine Learning, URL feature extraction, Naive Bayes, Phishing website.

I. INTRODUCTION

In recent years, phishing websites have become a major cyber security issue. Spam, malware, ransomware, drive-by vulnerabilities, and other malicious software can be found on phishing websites. A phishing website can sometimes be mistaken for a well-known website, luring an unsuspecting user into the trap. The victim of the fraud suffers a monetary loss, as well as the loss of personal information and reputation. As a result, it's critical to identify a solution that can quickly neutralize such security vulnerabilities. Blacklists have traditionally been used to detect phishing websites. Many notable websites, such as PhisTank, host a list of blacklisted domains. The blacklisting technique has two flaws: it may not be complete, and it does not detect newly created phishing attempts. . Machine learning techniques have been utilized to classify and detect phishing websites in recent years.

Global cyber security concerns have grown as a result of the evolving digital transformation. Cybercriminals have more opportunities as a result of digitization. To steal confidential user credentials, cyber dangers first approach in the form of phishing. Typically, hackers will try to sway users. Cybercriminals use security breaches to launch ransomware attacks, gain illegal access, shut down systems, and even demand a payment to regain access. Phishing software and techniques are used to get around anti-phishing software and techniques. . Despite the fact that threat intelligence and behavioral analytics technologies assist organizations in detecting anomalous traffic patterns, defending in depth remains the best technique for preventing phishing assaults. In this regard, the proposed research effort has built a model that uses machine learning (ML) methods such as random forest (RF) and decision tree to detect phishing assaults (DT). For ML processing, Kaggle provided a standard valid dataset of phishing attacks. The suggested model uses feature selection procedures like principal component analysis to analyse the dataset's properties (PCA).

Phishing assaults have become a major source of concern in the online community. It has a significant impact on internet users' privacy and financial matters. Scammers, namely fishermen, develop fake websites to defraud users by making them seem and look real. They send phoney emails to steal legitimate users' identities. They collect personal information, passwords, account information, and credit card information for the purpose of committing fraud. Fishermen's strategies for attacking the system are always changing. One of the most important strategies used by anglers is social engineering. They collect personal information from a reliable source using this method. Phishers develop fake websites and spoof emails that are remarkably similar to legitimate firm websites and sometimes look like they came from a source. Attackers may impersonate a legitimate source and push users to upgrade their systems. Furthermore, they threaten to suspend the customer's account and demand a ransom. Another tactic employed in phishing fraud is email spoofing. Customers are frequently duped into divulging personal information such as passwords and credit card numbers. As a result, fishing is mostly used to acquire sensitive data such as bank account, password, and credit card information. This type of scam is on the rise, and individuals and businesspeople are losing faith in internet transactions. As a result, clients developed a poor impression of Internet Company and lost faith in online transactions. Despite the fact that encryption software is used to safeguard the information stored on the computers, they are still vulnerable to attacks. In this paper, machine learning was used to detect fishing.

II. RELATED WORK

Phishing attacks are used to deceive consumers in a variety of ways. To defend against phishing assaults, several phishing detection techniques and technologies are now available. One of the strategies used to detect online phishing is classification. The following are examples of frequent phishing attacks and classification techniques:
A Phishing assaults of various types Attackers utilise a variety of tactics to test the security of the internet. They are constantly on the lookout for security system flaws to exploit. The following are examples of various phishing attacks that are distinct from one another.

1. Phishing based on algorithms America Online (AOL), which was built using an algorithm, was the first to catch a phishing attack. The fraudster used an algorithm to match the credit card numbers of America Online users.
2. Phishing as a Scam to deceive online users, the deceiver employs a variety of tactics. Fishers send account verification emails to users. There is a website behind the links where hackers steal and store the personal information of users.
3. Phishing of web addresses Phishing employing a disguised link is known as URL phishing. The hackers' website is accessible through the link. When a user hits the link, the user is transported to the hackers' website, where the information is stored.
4. In the Windows operating system platform, it's used to poison the host file. When a user finds the targeted website, it is redirected to a hacker's site, or the user receives an error message that says "The Page Not Found." Users' info is captured and stolen if it may redirect to a fraudulent website. 5)Phishing through the use of content injection Hackers prey on users by posing as a legitimate website.

The goal is to deceive the user or portray the company incorrectly. Content spoofing is another term for this. This approach is employed by the attackers in order to deceive the user and capture data on their server.
Due to the randomness of data themes, the current method for analysing valuable data in the world of Big Data delivers sentimental analysis on product reviews, which narrows the field of data exploitation and does not allow for the exploration of unstructured sources of information.
Other seemingly random data, particularly in academia, is organised based on tags. Manuals or specialists in their field of interest may or may not be aware of current trends, and they rely significantly on word of mouth to keep up with all the newest news and research in their field.

III.PROPOSED SYSTEM

A domain name is a registered name that is unique across the Internet and is registered with a We attempted to create a phishing detection system in this article by analysing the URL of the webpage. A URL is a long string that expresses syntactically and semantically the location of a resource on the Internet. Figure shows the structure of the URL when viewed in detail.

A. *Sets of data*: Phistank.com is a website that detects phishing URLs and may be accessed via an API request. Yahoo Mail, McAfee, APWG, Mozilla, Opera, Kaspersky, and Avira are just a few of the companies that use its data. The phishing data used in the machine learning method is often acquired from Phistank.com, according to the literature review. It completed the necessary categorisation of the previous URL addresses. It also included information on positive and negative (phishing/non-phishing) classifications. . It does not, however, preserve the content of webpages; as a result, it is an excellent source for URL-based analysis. In this paper, open-source and freely available datasets are employed. We chose open datasets for our comparative analysis. In this work, three datasets were employed, and the researchers termed their system Catch Phish. The first piece of data includes legitimate sites from Alexa and phishing sites from PhishTank. Second, legitimate and phishing sites from common-crawl and PhishTank, respectively. Third, legitimate sites were found in both the common-crawl and Alexa databases, while phishing sites were found in the PhishTank database.

B. *Extraction of Features:* The used features and machine learning methods have a direct impact on the efficacy of the taught system. As a result, we conducted a thorough literature analysis to identify the important elements. Studies that use elements from several categories, such as e-mail content analysis and website analysis, were also analysed in addition to those that merely analyse the URL. In the hostname, domain, and path sections, the features of URL were explored independently. We discovered 58 different aspects on this material during our research. Scripts built in the Python programming language were used to create these features. It was determined to use the best 48 characteristics after sorting them with the Random Forest Classifier. As a result, a better accuracy rate is desired. Table II lists the features that were used in the study. The tokens in the URL were obtained using the characters "/", ":", ".", "?", "=", "&", "-". Not every word mentioned in these tokens has been independently discovered. The token as a whole has been considered. In feature 12, the most common tokens were found and used. It was tested whether the top 10 TLDs of Alexa rank 1 million data were in the URL for feature 25. The system is built to classify URLs in a short amount of time utilising 48 features, without the use of third-party services or content analysis.

- **Using the IP Address**

Users can be confident that someone is attempting to steal their personal information if an IP address is used as a substitute to the domain name in the URL, such as
"http://125.98.3.123/fake.html."

$$\underline{Rule}\text{: IF}\begin{cases}\text{If The Domain Part has an IP Address} \rightarrow \text{Phishing}\\ \qquad\qquad \text{Otherwise} \rightarrow \text{Legitimate}\end{cases}$$

- **Long URL to Hide the Suspicious Part**

Phishers can disguise the suspect component of a URL in the address bar by using a lengthy URL. To verify the accuracy of our research, we computed the length of URLs in the dataset and created an Excel spreadsheet. Length of an average URL The findings revealed that if the URL length is higher than or equivalent to 54 characters, the URL is considered long. If the URL contains any of these characters, it is considered phishing. We discovered 1220 records while reviewing our database. URLs with lengths of 54 or more make up 48.8% of the overall dataset size.
We were able to improve the accuracy of this feature rule by updating it using a frequency-based technique.

$$\textit{Rule: IF}\begin{cases} \textit{URL length} < 54 \rightarrow \textit{feature} = \text{Legitimate}\\ \textit{else if URL length} \geq 54 \textit{ and } \leq 75 \rightarrow \textit{feature} = \textit{Suspicious}\\ \textit{otherwise} \rightarrow \textit{feature} = \text{Phishing}\end{cases}$$

- **Using URL Shortening Services "TinyURL"**

URL shortening is a way of reducing the length of a URL while still directing the user to the desired webpage on the "World Wide Web." This is done by using a "HTTP Redirect" on a short domain name that points to a large URL webpage.
For the URL "http://portal.hud.ac.uk/" can be abbreviated to "bit.ly/19DXSk4" as an example.

$$\underline{Rule}\text{: IF}\begin{cases}\text{TinyURL} \rightarrow \text{Phishing}\\ \text{Otherwise} \rightarrow \text{Legitimate}\end{cases}$$

- **URL's having "@" Symbol**

When you use the "@" sign in a URL, the browser ignores anything before the "@" symbol, and the true address usually comes after the "@" symbol.

$$\text{Rule: IF}\begin{cases}\text{Url Having @ Symbol} \rightarrow \text{Phishing}\\ \qquad \text{Otherwise} \rightarrow \text{Legitimate}\end{cases}$$

- **Redirecting using "//"**

If there is a "//" in the URL path, the user will be routed to another website. http://www.legitimate.com//http://www.phishing.com is an example of a URL like this. We look for the "//" in the right place. When a URL begins with "HTTP," we know that the "//" is present should be positioned sixth. If the URL uses "HTTPS," the "//" should be used instead occupy the eighth spot.

$$\text{Rule: IF}\begin{cases}\text{ThePosition of the Last Occurrence of "//" in the URL} > 7 \rightarrow \text{Phishing}\\ \text{Otherwise} \rightarrow \text{Legitimate}\end{cases}$$

- **Adding Prefix or Suffix Separated by (-) to the Domain**

In genuine URLs, the dash symbol is almost never used. Phishers often append prefixes or suffixes to the domain name, separated by (-), to give the impression that they are dealing with a legitimate website. Consider the website http://www.Confirme-paypal.com/.

$$\text{Rule: IF}\begin{cases}\text{Domain Name Part Includes } (-) \text{ Symbol} \rightarrow \text{Phishing}\\ \text{Otherwise} \rightarrow \text{Legitimate}\end{cases}$$

- **Sub Domain and Multi Sub Domains**

The country-code top-level domains (ccTLDs), such as “uk” in our example, may be included in a domain name. The “ac” component stands for “academic,” the combined “ac.uk” is known as a second-level domain (SLD), and “hud” stands for “high-level domain.”the domain's official name We must first exclude this characteristic in order to create a rule for extracting it.the (www.) in the URL, which is a subdomain in and of itself Then we must get rid of the if a ccTLD exists, use it. Finally, we add up all of the dots. If there are more than one dot, you've got a problem.Because it has one subdomain, the URL is regarded as "suspicious." If, on the other hand, the dots aren't connected, If there are more than two dots, it is considered "Phishing" since it will have several subdomains. If the URL does not have any subdomains, we will mark the feature as "Legitimate."

$$\text{Rule: IF}\begin{cases}\text{Dots In Domain Part} = 1 \rightarrow \text{Legitimate}\\ \text{Dots In Domain Part} = 2 \rightarrow \text{Suspicious}\\ \text{Otherwise} \rightarrow \text{Phishing}\end{cases}$$

- **HTTPS (Hyper Text Transfer Protocol with Secure Sockets Layer)**

The presence of HTTPS is critical in conveying the authenticity of a website, but it is by no means sufficient.
Furthermore, by putting our datasets through their paces, we've discovered that the bare minimum is two years is the minimum age for a credible certificate.

$$\text{Rule: IF}\begin{cases}\text{Use https and Issuer Is Trusted and Age of Certificate} \geq 1 \text{ Years} \rightarrow \text{Legitimate}\\ \text{Using https and Issuer Is Not Trusted} \rightarrow \text{Suspicious}\\ \text{Otherwise} \rightarrow \text{Phishing}\end{cases}$$

- **Domain Registration Length**

We assume that trustworthy domains are often paid for several years in advance, based on the fact that a phishing website only exists for a brief time. The longest bogus domains in our database were only used for one year.

$$\text{Rule: IF}\begin{cases}\text{Domains Expires on} \leq 1 \text{ years} \rightarrow \text{Phishing}\\ \text{Otherwise} \rightarrow \text{Legitimate}\end{cases}$$

- **Favicon**

A favicon is a graphic image (icon) that is connected with a specific webpage. Many existing user agents, such as graphical browsers and newsreaders, display the favicon in the address bar as a visual reminder of the website's identity. If the favicon is loaded from a domain other than the one listed in the address bar, it will appear in the address bar then the website is most likely a Phishing attempt.

$$\text{Rule: IF}\begin{cases}\text{Favicon Loaded From External Domain} \rightarrow \text{Phishing}\\ \text{Otherwise} \rightarrow \text{Legitimate}\end{cases}$$

- **Using Non-Standard Port**

This functionality is useful for determining whether a specific service (e.g. HTTP) is available on a specific server. It is much better to only open ports that you require in order to control incursions. Several firewalls, proxy servers, and network address translation (NAT) servers will block all or part of the traffic by default. Most of the ports are closed, and only the ones that are selected are opened. Phishers can run nearly any program if all ports are open. As a result, user information is jeopardised in order to provide them with the service they desire.

$$\text{Rule: IF}\begin{cases}\text{Port \# is of the Preffered Status} \rightarrow \text{Phishing} \\ \text{Otherwise} \rightarrow \text{Legitimate}\end{cases}$$

C.*System Implementation*: The machine learning-based system was tested using Bernoulli Naïve Bayes algorithm. The models developed with this algorithms was trained using the Python computer language module. In nonlinear problems, it can give effective solutions by employing the core trick. It is not, however, well suited to huge databases. It can't perform well if there's a lot of noise.

IV. SYSTEM DESIGN

The overall approach is depicted in Figure. The classifier's training phase accepts a labelled dataset of phishing and non-phishing URLs and outputs a trained model. For new instances, the trained model is utilised to make predictions.

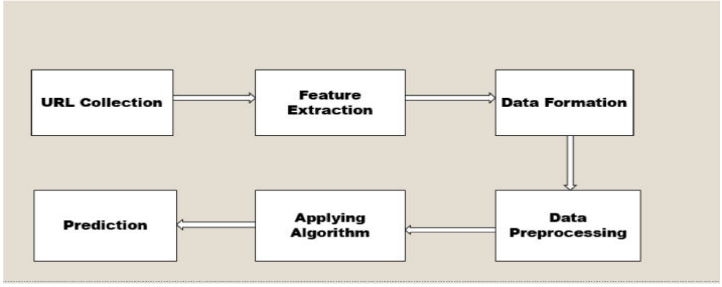


Fig. 1: System Architecture

IV.CONCLUSION AND FUTURE WORK

In this article, we looked at how effectively a set of URLs combining benign and phishing URLs might be used to classify phishing URLs. The randomization of the dataset, feature engineering, feature extraction utilising lexical analysis host-based features, and statistical analysis have all been explored.For the comparative analysis, we tested a variety of classifiers and discovered that the results are nearly identical across all of them. We also discovered that randomising the dataset resulted in a significant improvement in the classifier's accuracy. Using simple regular expressions, we extracted the features from the URLs in a straightforward manner. There may be more characteristics that can be tested, which could lead to the system's accuracy being improved even further. The URLs list in the dataset used in this paper may be a little old, thus regular continuous training with a new dataset would considerably improve model accuracy and performance. We did not utilise content-based features in our experiment since the fundamental challenge with using content-based features to detect phishing URLs is that phishing websites are rarely available, and their lifespan is short, making it impossible to train an ML classifier using content-based features. We'd like to include a rule-based prediction based on a URL's content analysis in the future. As a result, a full solution for phishing URL detection would be a combination of a classification-based lexical analyser and a rule-based URL content analyzer.

REFERENCES

1. Samuel Marchal, Jérôme François, Radu State, and Thomas Engel, "PhishStorm: Detecting Phishing With Streaming Analytics," IEEE Transactions on Network and Service Management, vol. 11 , issue: 4 , pp. 458-471, December 2014.
2. Mohammed Nazim Feroz,Susan Mengel, "Phishing URL Detection Using URL Ranking," IEEE International Congress on Big Data, July 2015.
3. Mahdieh Zabihimayvan, Derek Doran, "Fuzzy Rough Set Feature Selection to Enhance Phishing Attack Detection," International Conference on Fuzzy Systems (FUZZ-IEEE), New Orleans, LA, USA, June 2019.
4. Moitrayee Chatterjee,Akbar-Siami Namin, "Detecting Phishing Websites through Deep Reinforcement Learning," IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC), July 2019.
5. Chun-Ying Huang,Shang-Pin Ma,Wei-Lin Yeh,Chia-Yi Lin,ChienTsung Liu, "Mitigate web phishing using site signatures," TENCON 2010-2010 IEEE Region 10 Conference, January 2011.
6. Aaron Blum,Brad Wardman,Thamar Solorio,Gary Warner, "Lexical feature based phishing URL detection using online learning," 3rd ACM workshop on Artificial intelligence and security, Chicago, Illinois, USA, pp. 54-60,

August 2010. 7. Mohammed Al-Janabi,Ed de Quincey,Peter Andras, "Using supervised machine learning algorithms to detect suspicious URLs in online social networks," IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017, Sydney, Australia, pp. 1104-1111, July 2010.

7. Erzhou Zhu,Yuyang Chen,Chengcheng Ye,Xuejun Li,Feng Liu, "OFSNN:An Effective Phishing Websites Detection Model Based on Optimal Feature Selection and Neural Network," IEEE Access(Volume:7), pp. 73271-73284, June 2010.
8. Ankesh Anand,Kshitij Gorde,Joel Ruben Antony Moniz,Noseong Park,Tanmoy Chakraborty,Bei-Tseng Chu, "Phishing URL Detection with Oversampling based on Text Generative Adversarial Networks," IEEE International Conference on Big Data (Big Data), December 2018.
9. Justin Ma,Lawrence K. Saul,Stefan Savage,Geoffrey M. Voelker, "Learning to detect malicious URLs," ACM Transactions on Intelligent Systems and Technology (TIST) archive Volume 2 Issue 3, April 2011

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING