International Journal for Multidisciplinary Research (IJFMR)

E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Detection of Phishing URLs Using a Term Frequency Inverse Document Frequency (TF-IDF)

Sibhathallah M[1], Dr. D Sathya Srinivas[2]

[1]M.Sc., Department of Computer Science and Engineering, Dr. MGR Educational and Research Institute, Chennai, India
[2]Faculty, Centre of Excellence in Digital Forensics, Dr. MGR Educational and Research Institute, Chennai, India.

Abstract:
Phishing attacks continue to pose a threat to online security as hackers employ more sophisticated tactics to trick users into revealing sensitive information. One common method is creating fake login URLs that mimic legitimate websites, making it challenging for users to distinguish between safe and dangerous links. This study focuses on detecting real-world phishing URLs, particularly analyzing login URLs. Our aim is to develop effective techniques and tools for early detection and prevention of phishing attempts by examining the common traits and patterns associated with them. The paper seeks to enhance people's and organizations' resilience to phishing attacks by exploring advanced technology and machine learning algorithms, ultimately contributing to a safer online environment. The research evaluates the effectiveness of deep learning and machine learning techniques in classifying phishing URLs, using statistical features such as Term Frequency-Inverse Document Frequency (TF-IDF) in combination with character N-gram, as well as proposed handcrafted features. CNN models are utilized for the deep learning approaches. Following model training, it is used to classify phishing URLs. The primary aim of this experiment is to assess the effectiveness of our proposed method. The results demonstrate that our phishing URL detection method achieves an accuracy rate of 96.6%

Keywords: Cybercrime, Machine Learning, URLs, Prediction, Phishing Detection.

Introduction:
Malicious URLs are a common way for cyber criminals to carry out scams, such as identity theft, malware installation, and stealing money. This results in billions of dollars of losses every year. To detect malicious URLs, blacklisting methods are often used, which involves using a list of known malicious URLs [1]. However, this method is not foolproof, and new URLs that are generated every day can slip through the cracks. Machine learning models are becoming more common in detecting malicious URLs, as they can generalize their predictions to unseen URLs. The first step in this process is obtaining a feature representation of the URL, which can be done by using lexical features, host-based features, content features, context features, or popularity features. The second step is to use this feature representation to train machine learning models, such as SVMs. However, these approaches have limitations, such as not being able to capture semantic or sequential patterns effectively, requiring expert guidance for feature

engineering, and being unable to handle unseen features. to address these issues, URL net, a deep learning-based solution for malicious URL detection, has been proposed. URL Net uses convolutional neural networks (CNNs) to learn a URL embedding for malicious URL detection. It receives a URL string as input and applies CNNs to both characters and words in the URL. Character-level CNNs identify important information from groups of characters that could indicate maliciousness, while word-level CNNs identify useful patterns from groups of words.

The rapid development of communication technologies has led to a shift in our daily life activities, such as social networks, electronic banking, e-commerce, etc., [2]to the cyberspace. However, the open and uncontrolled infrastructure of the internet has created a platform for cyberattacks, making networks and computer users vulnerable to security issues. Even experienced users can fall prey to phishing scams, as attackers consider their personality characteristics to increase the success of their attacks. End-user-targeted cyberattacks can result in significant losses of sensitive personal information and money, totaling billions of dollars annually. Phishing attacks involve creating fraudulent websites that have the same design as popular and legal sites on the internet. Although these pages may have similar graphical user interfaces, they will have different Uniform Resource Locators (URLs) from the original page. A careful and experienced user can easily detect these malicious web pages by examining the URLs. However, due to the fast-paced nature of life, end users often fail to investigate the entire address of their active web page, which is usually forwarded by other web pages, social networking tools, or email messages. Using fraudulent URLs, phishers capture sensitive personal information from their victims, such as financial data, personal information, username, password, etc. Victims are easily fooled into giving away their sensitive information because the fraudulent website appears identical to the original one.

In today's world, digital platforms play a crucial role in our daily routines. Using computers and the internet has made our lives easier and more efficient in various fields, including trade, health, education, communication, banking, aviation, research, engineering, entertainment, and public services. With the advancement of mobile and wireless technologies, users can now easily connect to the internet from anywhere at any time. However, this convenience has also resulted in serious information security issues. As a result, users in cyberspace must take measures against possible cyber-attacks, which can be carried out by cybercriminals, pirates, or non-malicious (white-capped) attackers and hacktivists. The attackers aim to obtain access to the computer or the information it contains or to capture personal information in different ways, such as fraud, forgery, force, shakedown, hacking, service blocking, malware applications, illegal digital content, and social engineering. These attacks have been carried out since 1988 and continue to this day. According to Kaspersky's data, the cost of an attack in 2019 ranged from According to Kaspersky's data, the cost of an attack in 2019 ranged from $108K to $1.4 billion [3] depending on the scope of the attack.4 billion, depending on the scope of the attack. Furthermore, the global expenditure on security products and services is approximately Furthermore, the global expenditure on security products and services is approximately $124 billion.24 billion [4]. The most widespread and critical type of attack is "phishing attacks," in which cybercriminals use email or other social networking communication channels to give victims the impression that the post was sent from a reliable source, such as a bank or e-commerce site. The attackers then attempt to gain sensitive information from their victims, which they use to access their accounts, causing pecuniary loss and intangible damages.

The drawbacks of blacklist-based methods have led researchers to focus on using machine learning for automatic detection of phishing URLs [5], [6]. These approaches can be categorized based on the type of data used: URL text, page content, visual features, and networking information [7]. Methods that rely on page content and visual features require visiting the website to collect and render the source code, making them time-consuming. Limitations can be found in studies relying on networking and third-party information, such as WHOIS or search engine rankings. To overcome these limitations, we are concentrating on phishing detection through URLs, which offers advantages such as fast computation (as no websites are loaded) and being third-party and language independent, as features are extracted only from the URLs.

Review Of Literature:

Asadullah Safi, Satwinder Singh, and et al., [8] proposed that phishing is a fraudulent attempt where an attacker pretends to be a trustworthy entity to obtain sensitive information from an internet user. In their Systematic Literature Survey (SLR), they studied and compared various techniques for detecting phishing, such as Lists Based, Visual Similarity, Heuristic, Machine Learning, and Deep Learning approaches. The researchers examined multiple algorithms, datasets, and techniques used for detecting phishing websites and proposed research questions. The SLR analyzed 80 scientific papers published in the last five years in research journals, conferences, leading workshops, theses, book chapters, and high-ranking websites. The study is an update to previous systematic literature surveys, focusing on the latest trends in phishing detection techniques. It enhances understanding of various phishing website detection methods, datasets used, and algorithm performance. The SLR found that Machine Learning techniques have been the most widely applied, with 57 studies using them. Moreover, researchers primarily accessed two sources while gathering datasets: 53 studies accessed the Phish Tank website, and 29 studies used Alexa's website for downloading legitimate datasets. According to the literature survey, most of the studies utilized Machine Learning techniques, with 31 employing the Random Forest Classifier. Various studies have found that the Convolutional Neural Network (CNN) achieved a 99.98% accuracy in detecting phishing websites.

Marwa Al Saedi, Nahla Abbas Flayh, and et al.,[9] proposed that phishing, a form of cyberattack in which perpetrators use fraudulent websites or emails to trick individuals into revealing sensitive information such as passwords or financial data, can be mitigated in a variety of ways. Various machine learning algorithms can be used for website detection. These algorithms, including Decision Trees, Support Vector Machines, and Random Forests, analyze several characteristics of a website, such as URL structure, website content, and the presence of keywords or patterns, to determine the likelihood that the page is a fraudulent site. This review provides an overview of phishing website detection, including techniques and summarizing previous studies, findings, and contributions. Machine learning algorithms detect phishing websites to protect users from malicious attacks

Kang Leng Chiew, [10] proposed that anti-phishing research is an important field in information security. Unfortunately, there is no standard test dataset available for researchers to use. This means that most researchers must create their own datasets for their experiments, making it difficult and inefficient to compare different anti-phishing techniques. To address this problem, a large-scale offline dataset has been created. It is comprehensive, universal, and available for download. The dataset creation approach considered the requirements of major anti-phishing techniques from existing literature. Several factors were identified as important for enhancing the dataset quality, including the type of raw elements, the source of the sample, sample size, website category, category distribution, language of the website, and support for feature extraction. The dataset construction approach resulted in 30,000 samples, with an equal distribution of 50% for phishing and legitimate webpages. This dataset can be used for anti-phishing research, benchmarking, and rapid proof-of-concept experiments. Given the rapid evolution of phishing attacks, the need for such a dataset cannot be overstated.

Eduardo Fidalgo, [11] proposed that phishing attacks are complex cyberthreats that involve deceiving users to steal their credentials or sensitive information. They developed a method for detecting phishing sites using URL, HTML, and web technology features. They created the Phishing File Login Websites Dataset (PILWD) comprising 134,000 samples. Their approach achieved 97.95% accuracy in detecting phishing websites using a Light GBM classifier and a set of 54 features.

Abhijit Sarma, [12] proposed that phishing is an illegitimate method used by attackers to collect secret information of individuals or organizations, such as debit card and credit card details, PIN numbers, OTPs, passwords, etc. Researchers have developed various techniques to detect and identify phishing websites, but it can be challenging to rely on a particular technique as attackers often come up with new tactics. The paper suggests a method of classifying phishing and legitimate URLs based on the lexical features of URLs. A feature selection technique is used to select only relevant features to improve the accuracy of the classification. The performance of different combinations of features is evaluated to find the best possible combination using various datasets and parameters. Four different machine learning techniques are used to analyze the performances

V. V. Ramalingam, Paras Yadav, and et al., [13] proposed that phishing is a type of cyber-attack that tricks people into giving away their sensitive information, such as login credentials, social security numbers, and banking details. Attackers do this by pretending to be a legitimate webpage or reputable company through emails. They designed a model that extracts heuristic features from website domains, URLs, web protocols, and source codes. This model uses blacklisting, whitelisting, heuristics, and visual-based similarity to boost security. They then used this model with different Machine Learning Algorithms and compared the results to find the most efficient framework.

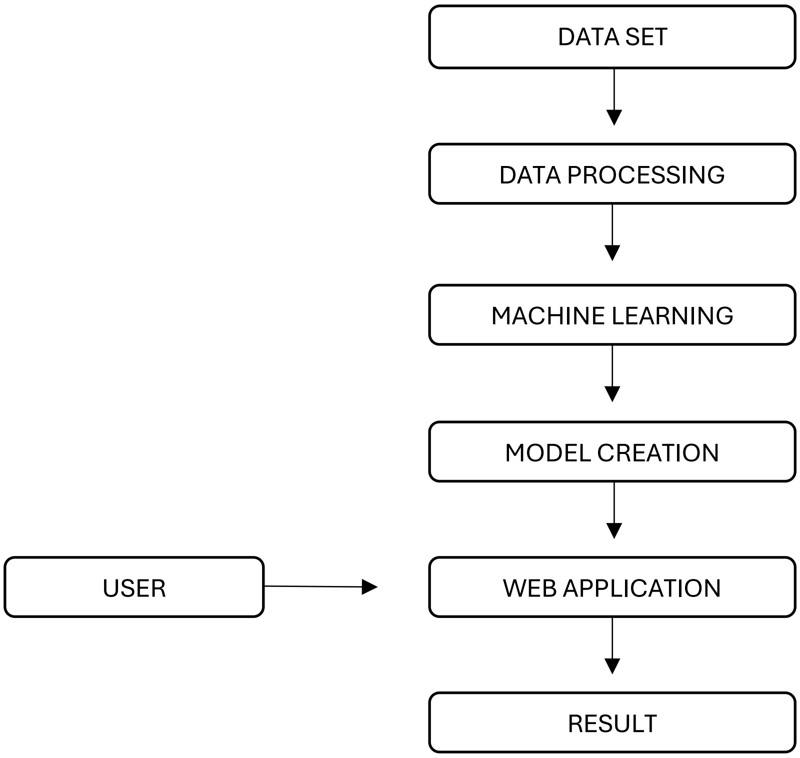
Mehmet Korkmaz, [14] have proposed that with the increasing use of mobile devices in recent years, there is a growing trend to move almost all real-world operations to the cyberworld. While this makes our daily lives easier, it also brings many security breaches due to the anonymous structure of the Internet. Most attacks can be prevented by antivirus programs and firewall systems, but experienced attackers often target the weaknesses of computer users by attempting to phish them with fake webpages. These pages imitate popular banking, social media, e-commerce, etc. sites to steal sensitive information such as user IDs, passwords, bank account and credit card numbers. The market provides solutions for the difficult challenge of phishing detection, including blacklists, rule-based detection, and anomaly-based detection. In the literature, it is seen that current works tend to use machine learning-based anomaly detection due to its dynamic structure, especially for catching "zero-day" attacks. Inorder to compare the findings with previous research, this study proposes a machine learning-based phishing detection system that analyzes URLs using eight distinct methods and three different datasets. The experimental results depict that the proposed models have outstanding performance with a high success rate.

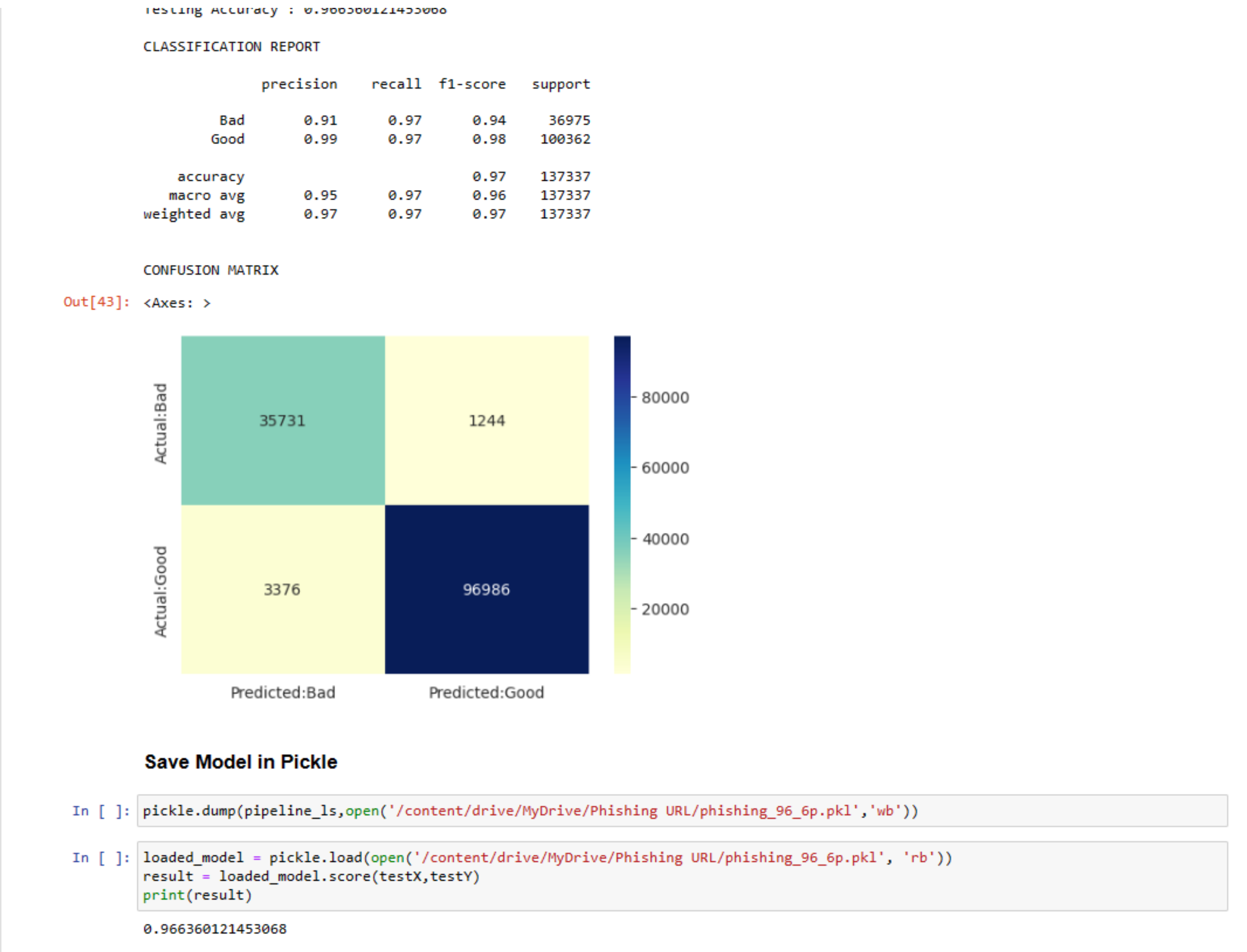
Research Methodology:

To use the methodology outlined in this research, you will need a laptop or computer with Anaconda Server. Start by turning on the computer, then open a browser and search for Anaconda Server [15] to download and install it on your system. Once installed, open Anaconda Server and run it, and use Jupyter within it. Next, locate the project file, such as modules and coding created by us, and search for Anaconda Prompt in the search taskbar, then open it. In Anaconda Prompt, use the change directory command to locate the module pages where that file is located. Enter Python commands to provide a web page link, then copy that link and paste it on web servers like Chrome or Microsoft Bing. The created web page will open. In that page, the prediction bar will be located. In that bar, paste the URL that was doubted by you and press the predict button. The page will predict that URL as good or bad.

In terms of machine learning techniques, I utilized (1) statistical features using Term Frequency-Inverse Document Frequency (TF-IDF) in conjunction with character N-gram and (2) handcrafted features. I used CNN models for the deep learning approaches [16].

In the backend, we used machine learning and deep learning with the help of Python. I imported NumPy, Pandas, Matplotlib.pyplot, and seaborn. In my proposed system, I collected real-time data from the Kaggle website. The data was collected manually and stored as a dataset, which will be used for training to achieve high accuracy results. Then, I loaded the dataset from the database. I used natural language processing with the NLTK tool, including tokenization with regex tokenizer to remove unwanted symbols and numbers, as well as a stop word removal process. Finally, i stemmed the words using the Snowball Stemmer from the NLTK tool. Using Regexp Tokenizer, I tokenized the given datasets. The tokenizer splits the URLs separately, and then removes [.,';"] from the URL. Next, I used Snowball Stemmer to string the words and separate URLs as Good and Bad. Then, I used Counter Vectorizer to convert all words to 0s and 1s, and then transform them to an array. I used the Logistic Regression algorithm [17] to train the URL, splitting it as X and Y. I used LR performance matrix to test the URLs (test x and test y) and classify the report as Good and Bad. Then, I used MNB Performance Matrix to train and test the accuracy. I then used logistic regression to connect all processes and perform pipeline performance matrix. Finally, i saved the models in pickle. From the model, results will be printed. Using the Naive Bayes algorithm [18], results will be predicted. The prediction will indicate whether the URL is good or bad. The result will show if the URL is malicious, in which case it will be labeled as a bad URL, or if it's non-malicious, it will be labeled as a good URL. This result will help identify whether the URL is malicious or non-malicious.





```
Testing Accuracy : 0.966360121453068

CLASSIFICATION REPORT

              precision    recall  f1-score   support

         Bad       0.91      0.97      0.94     36975
        Good       0.99      0.97      0.98    100362

    accuracy                           0.97    137337
   macro avg       0.95      0.97      0.96    137337
weighted avg       0.97      0.97      0.97    137337


CONFUSION MATRIX
```

```
Out[43]: <Axes: >
```

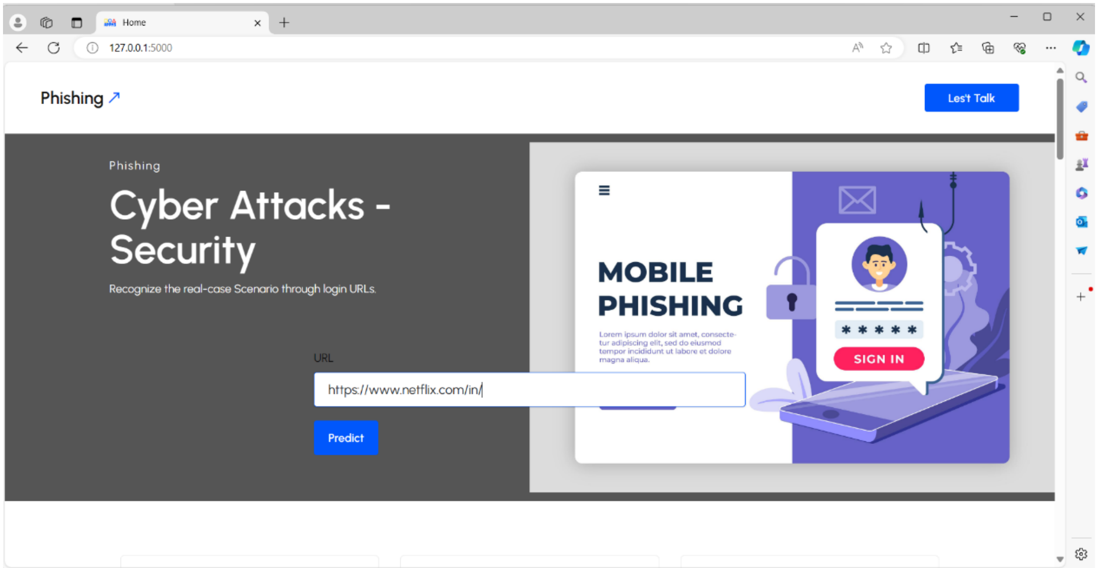
Save Model in Pickle

```
In [ ]: pickle.dump(pipeline_ls,open('/content/drive/MyDrive/Phishing URL/phishing_96_6p.pkl','wb'))
```

```
In [ ]: loaded_model = pickle.load(open('/content/drive/MyDrive/Phishing URL/phishing_96_6p.pkl', 'rb'))
        result = loaded_model.score(testX,testY)
        print(result)

        0.966360121453068
```

Fig-1: Final accuracy rate.



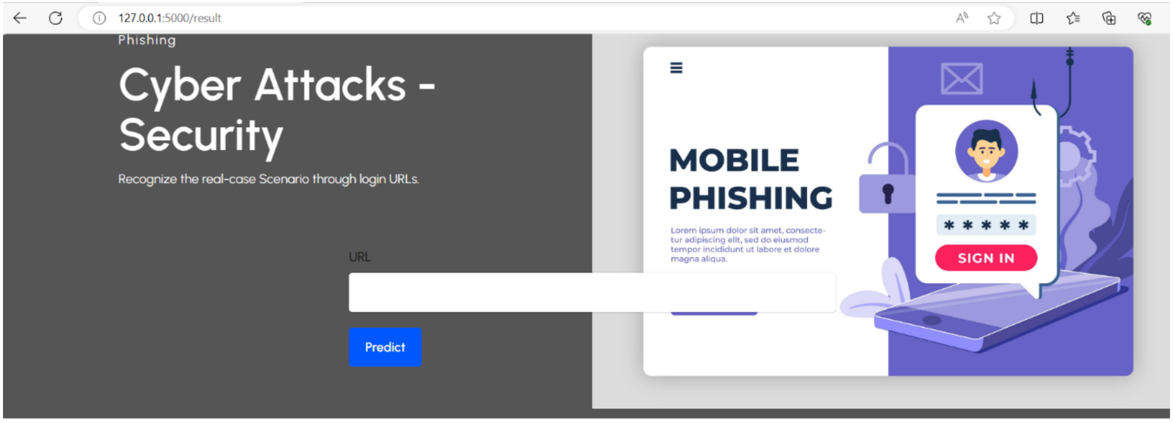
Home
127.0.0.1:5000
Phishing ↗
Lest Talk
Phishing

Cyber Attacks - Security

Recognize the real-case Scenario through login URLs.

URL: https://www.netflix.com/in/

Predict
MOBILE PHISHING
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.
SIGN IN



Phishing

Cyber Attacks - Security

Recognize the real-case Scenario through login URLs.

127.0.0.1:5000/result
URL
Predict
MOBILE PHISHING
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.
SIGN IN

Its a good URL

List Of Abbreviations:

TFIDF- Term Frequency Inverse Document Frequency
CNN- Convolutional Neural Network
URL-Uniform Resource Locator
SVMs-Support Vector Machines
SLR-Simple Linear Regression
HTML-Hypertext Markup Language
PILWD- Phishing File Login Websites Dataset
PIN- Personal Identification Number
OTP-One Time Password
NLTX-Natural Language Toolkit

Conclusion:

We have demonstrated that machine learning models using handcrafted URL features show improved performance. Therefore, it is important to train machine learning methods with recent URLs to prevent substantial aging from the date of release. The dataset includes legitimate login URLs, which are the most representative scenario for real-world phishing detection. We explored several URL-based detection models using deep learning and machine learning solutions trained with phishing and legitimate home URLs. The main advantage of our approach is the low false-positive rate when classifying this type of URL.

Reference:

1. R. De', N. Pandey and A. Pal, "Impact of digital surge during COVID-19 pandemic: A viewpoint on research and practice", *Int. J. Inf. Manage.*, vol. 55, Dec. 2020.
2. 2.Statista. (2020). Adoption Rate of Emerging Technologies in Organizations Worldwide as of 2020. Accessed: Sep. 12, 2021.
3. Phishing Activity Trends Report 4Q, Sep. 2020.
4. Gunter Ollmann, “The Phishing Guide Understanding & Preventing Phishing Attacks”, IBM Internet Security Systems, 2007.

5. L. Halgas, I. Agrafiotis and J. R. C. Nurse, "Catching the phish: Detecting phishing attacks using recurrent neural networks (RNNs)" in Information Security Applications, Cham, Switzerland: Springer, vol. 11897, pp. 219-233, 2020
6. R. S. Rao and A. R. Pais, "Jail-phish: An improved search engine-based phishing detection system", *Comput. Secur.*, vol. 83, pp. 246-267, Jun. 2019.
7. Z. Dou, I. Khalil, A. Khreishah, A. Al-Fuqaha and M. Guizani, "Systematization of knowledge (Sok): A systematic review of software-based web phishing detection", *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2797-2819, 4th Quart.
8. Asadullah Safi, Satwinder Singh, A systematic literature review on phishing website detection techniques, Journal of King Saud University -Computer and Information Sciences, Volume 35, Issue 2,2023, Pages 590-611.
9. saedi, Marwa & Flayh, Nahla. (2023). Phishing Website Detection Using Machine Learning: A Review. Wasit Journal for Pure sciences.270-281. 10.31185/wjps.145.
10. Chiew, Kang Leng & Chang, Ee & Tan, Colin Choon Lin & Abdullah, Johari & Yong, Kelvin. (2018). Building Standard Offline Anti-Phishing Dataset for Benchmarking. 7-14. 10.14419.
11. Manuel Sánchez-Paniagua, Eduardo Fidalgo, Enrique Alegre, Rocío Aliaz-Rodríguez, Phishing websites detection using a novel multipurpose dataset and web technologies features, Expert Systems with Applications Volume 207,2022.
12. Banik, Bireswar & Sarma, Abhijit. (2020). Lexical Feature Based Feature Selection and Phishing URL Classification Using Machine Learning Techniques. 10.1007/978-981-15-6318-8_9.
13. Routhu, Srinivas & Pais, Alwyn. (2019). Detection of phishing websites using an efficient feature-based machine learning framework. Neural Computing and Applications. 31. 10.1007/s00521-017-3305-0.
14. Korkmaz, Mehmet & Sahingoz, Ozgur & Diri, Banu. (2020). Detection of Phishing Websites by Using Machine Learning-Based URL Analysis. 1-7. 10.1109/ICCCNT49239.2020.9225561.
15. Anaconda Software Distribution. (2020). Anaconda Documentation. Anaconda Inc. Retrieved from https://docs.anaconda.com/
16. Sanchez Paniagua, Manuel & Fidalgo, Eduardo & Alegre, Enrique & Wesam, Al-Nabki & González-Castro, Víctor. (2022). Phishing URL detection: A real-case scenario through login URLs. IEEE Access. 10. 1-1. 10.1109/ACCESS.2022.3168681.
17. Maalouf, Maher. (2011). Logistic regression in data analysis: An overview. International Journal of Data Analysis Techniques and Strategies. 3. 281-299. 10.1504/IJDATS.2011.041335.
18. Lewis, D. (1998) Naive Bayes at forty: the independence assumption in information retrieval. In Machine Learning: ECML-98, Proceedings of the 10th European Conference on Machine Learning, Chemnitz, Germany (pp. 4–15). Berlin: Springer.