RESEARCH ARTICLE OPEN ACCESS

# Enhancing Cybersecurity Through Optical Character Recognition (OCR)

Samarth Math, Chetan Chaudhari, Dr. Sarika Jadhav

(MCA, D Y Patil International University, Pune
Email: samarthmath7@gmail.com)
(MCA, D Y Patil International University, Pune
Email:Chetanchaudhari588@gmail.com)
(MCA, D Y Patil International University, Pune
Email: Sarika.jadhav@dypiu.ac.in)

----------------------------------------************************----------------------------------

## Abstract:

Technological innovation is rapidly accelerating in a Cyber world that is powered by social networks, online transactions, cloud computing, and automated processes. The technology evolution often brings it with the advancement of cybercrime. Which leads to Evolve in Security tools, techniques, and attack types, allowing attackers to penetrate more complex, even while remaining undetected. To make our systems more secure, it is crucial to know about those attacks, before and after they occur. Cyber security experts suggest that it is hard to predict an attack without knowing how vulnerable a network is. Therefore, it is important to analyzea networkto determine top vulnerabilities, which can give the best idea of how to shield the network. It is the ever-evolving nature of Cyber Attacks that represents the main challenge of cyber security specialist. In this paper, we will discuss the importance of cyber security and the different risks that are present in the current security era. We will also evaluate countermeasures that can be implemented when attacks occur with the proposed process model of File Upload Vulnerability. This research also seeks to identify vulnerabilities through attacks and to provide mitigation for those vulnerabilities.

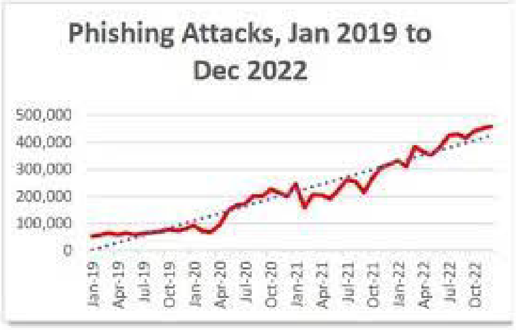
*Keywords* — OCR, Vulnerabilities, Cyber Security

----------------------------------------************************----------------------------------

## I. INTRODUCTION

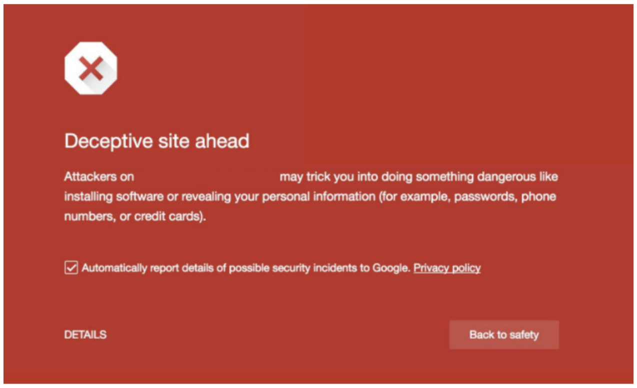
The internet has become an essential part of our daily lives, offering a wide range of resources for work, education, and entertainment. With the increasing use of mobile devices, people can access many services, including banking, at their fingertips. More and more consumers are managing their finances through mobile banking apps, as indicated by data from the British Bankers Association and Ernst & Young in 2017. However, mobile security remains a significant challenge. While companies like Apple and Google have security measures for devices and apps, they struggle to prevent phishing attacks. A survey by McAfee in 2015 found that around 97% of consumers couldn't identify phishing emails correctly. Additionally, the rise of QR codes has introduced QR phishing as a new threat. Unlike with computers, mobile devices and browsers often lack secure ways for users to recognize phishing URLs. This makes it difficult to distinguish phishing from legitimate websites.

For phishing emails and QR phishing, the malicious website addresses are often hidden, and many users don't check the website they are accessing. A survey by Wandera revealed that a new phishing site is created every 20 seconds. Listbased phishing protection services are not very effective at blocking these suspicious sites, as they can't detect and update the list in real-time.



## II. RELATED WORK

A study by Felt and Wagner [2], the research focused on assessing the risks associated with phishing on mobile platforms. One significant concept introduced in their work was the notion of control transfer. They emphasized the importance of mobile websites adhering to the standard Same Origin Policy [1] as a measure to isolate potentially untrustworthy websites from each other. However, it was noted that neither Android nor Apple imposed stringent restrictions on mobile website access. Consequently, phishing incidents often occurred during control transfers, such as trusted inter-application links, which could potentially lead users to malicious destinations. In the research conducted by Krombholz et al. [8], the study shed light on the emerging use of QR codes as a vector for phishing attacks. QR codes were highlighted as a costeffective and easily deployable means for attackers to carry out their malicious intentions. Attackers could either entirely replace a legitimate QR code or subtly alter a few pixels to transform it into a malicious vector. These malicious QR codes were designed to redirect users to phishing scams. Sharma's data [9] supported this by indicating that the first known malicious case using QR codes was detected in September 2011.


Deceptive site ahead

Attackers on may trick you into doing something dangerous like installing software or revealing your personal information (for example, passwords, phone numbers, or credit cards).

☑ Automatically report details of possible security incidents to Google. Privacy policy

Secondly, mobile platforms face phishing risks related to Quick Response (QR) codes. QR codes are encoded, rendering them unreadable by humans and requiring specific QR readers to decode the information. Consequently, it is plausible to trigger potential vulnerabilities, such as buffer overflows or command injections, by manipulating QR codes [17] [18]. Additionally, a study comparing 31 QR scanner applications revealed that only two apps featured a security warning feature, but they also exhibited a higher rate of false negative errors [19]. Subsequently, two more reliable opensource APIs, namely Google Safe Browsing and PhishTank, were recommended to enhance phishing prevention accuracy (True Positives). However, the static limitations of blacklists have yet to be fully overcome.

## III. METHODOLOGY

Research Aim: This study aims to determine the purpose of a website by analyzing its logo or background image through image recognition techniques. It then compares these visual elements to the official website to identify potential phishing activity based on the accessed URL. The implementation of this aim is divided into four steps as follows: i. Extract the image. ii. Describe the image content. iii. Confirm the official URL. iv. Verify the accessed URL. Preliminary Requirement: To test the feasibility of the initial methodology, a Python program was developed to detect images and search for the related official website. The implementation required the registration of certain opensource APIs, including: Google Optical Character Recognition (OCR) API. Google Search API.

Procedures: The research involved the following steps, which were applied to 40 URLs sourced from PhishTank for proof of concept:

1) Extract the image: Web crawling was used to retrieve related images from websites. This process considered two common methods of linking logo or background images: via HTML code and .css files. The HTML approach involved the logo image being inserted under the **Error! Filename not specified.** tag in the HTML code, while the .css approach used attributes like background or backgroundimg. Both methods were considered to account for the diverse nature of phishing.

2) Describe the image content: The Google Optical Character Recognition (OCR) API was used to identify textual content within images. This step excluded redundant images, such as symbol icons from .css files, scenery, or character images from HTML **Error! Filename not specified.** tags, to improve efficiency. Other OCR APIs, such as Microsoft Azure, were also tested, but their results were not as satisfactory

3) Confirm the official URL: Based on the image content description, the expected purpose of the website was determined. In this step, the related official URL addresses were obtained using the Google Search API through keyword searches. The top three results from the search were considered, with the first usually being the official URL, the second linking to a Wiki page about the website, and the third providing related news or other branches of the website.

4) Verify the Accessed URL: As most websites have implemented SSL (Secure Sockets Layer) to enhance data security through encrypted connections between web servers and browsers [20], we verify the security of the accessed URLs by comparing SSL certificate information. Initially, we attempted to retrieve the SSL certificate and its associated hash thumbprint to confirm the consistency between the accessed URLs and the official URL. However, the results did not meet our expectations, as hash values may vary under different domain names or branches within the same company. For instance, as illustrated in Figure 4, the SSL certificate hash values differ between https://www.google.com/ and https://www.google.co.uk/. As a result, we opted to utilize the organization name present in SSL certificates to ascertain that these websites are associated with the same company. This approach enabled us to verify the security of the accessed URL across all registered URLs.

## IV. EVALUATION

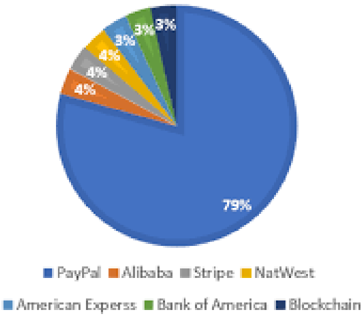
Analysis and Findings: To assess the effectiveness of this approach, a set of malicious URLs were identified from PhishTank, meeting the following criteria:

1) These URLs had to be accessible online during the detection phase.

2) The logos of these websites had to contain textualelements since an API was required to provide descriptions.

3) The logos were required to be in English, as the OCR API used in the prototype supported English text recognition.

In our initial testing, we collected 40 diverse phishing URLs from PhishTank randomly. Notably, a significant majority of these reported phishing URLs, approximately 72.5% (29 out of 40), pertained to financial matters, encompassing sites such as PayPal, Alibaba, American Express, NatWest, and others, as depicted in Figure 5. Within this financial category, PayPal accounted for the highest proportion at 79% (23 out of 29).



Within the framework of our approach, we achieved a success rate of 90% in identifying phishing URLs, successfully flagging 36 out of the 40 tested URLs. Only four URLs eluded detection. A comprehensive analysis of these instances of misidentification revealed that the primary contributing factor was related to the logo extraction phase. Essentially, when the logo image was extracted unclearly or inaccurately in the initial step, it led to an incorrect determination of the website's purpose in Step 2. Further scrutiny of the outcomes disclosed two specific reasons for these inaccuracies: 1) Text-Based Logos: In some instances, the website's logo was composed of text rather than a graphical image. This presented a challenge for accurate identification through the web crawler. 2) Complex Image-Based Layouts: For particular websites, the entire page was presented as a single image, containing an extensive amount of intricate detail. This complexity posed difficulties for the OCR API to process effectively.

## V. LIMITATIONS AND CHALLENGES

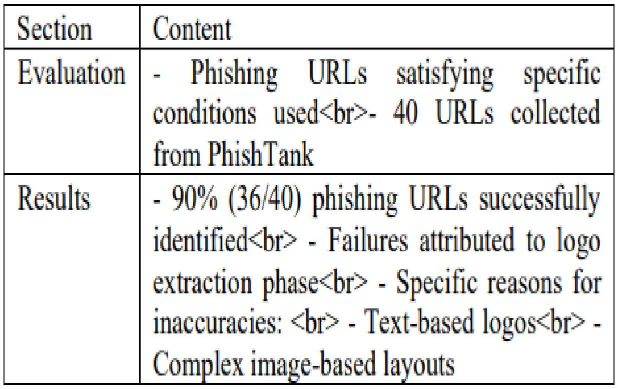
Undeniably, Optical Character Recognition (OCR) technology is a pivotal component of our approach, and the accuracy of the final results hinges on the quality of recognized details. Initially, we attempted to integrate Microsoft Azure's OCR API into our methodology. However, the outcomes fell short of our expectations. This was primarily due to the API's limitations, including its inability to perform effectively in specific scenarios, such as logos with dark backgrounds, and its support only for JPEG, PNG, GIF, and BMP image formats. Some websites employ SVG files for their logo images, which the Microsoft OCR API could not accommodate. Consequently, we transitioned to the Google OCR API for our implementation, as it overcame these prior limitations. The effectiveness of logo image extraction is heavily reliant on the web crawler's performance. The recognized results are rendered ineffective if the extracted logo image is inaccurate. In light of these considerations, the limitations and challenges can be summarized as follows:

1) Accuracy of Logo Image Extraction: In the diverse landscape of phishing, logos may be inserted using various methods. It can be challenging to locate a logo image when, for instance, the entire webpage is presented as a single image.

2) Cost of OCR API: While multiple APIs can be employed for text recognition from images, they are not offered free of charge. A daily free quota, typically around 1000 uses, is provided, but any additional checks incur charges. Therefore, the cost of utilizing OCR APIs must be taken into account for larger-scale processing.

3) System Efficiency: Phishing websites exhibit diversity and complexity. A phishing URL may contain numerous images or .css files. In such scenarios, both the web crawling and image recognition processes can become significantly more time-consuming.

4) Single Detectability: The threat posed by phishing extends beyond the theft of private information. Phishing websites can also execute viruses that infect victims, incorporating them into a botnet. However, this approach does not address the risk associated with implanted viruses on phishing sites.



| Section | Content |
|---|---|
| Evaluation | - Phishing URLs satisfying specific conditions used<br>- 40 URLs collected from PhishTank |
| Results | - 90% (36/40) phishing URLs successfully identified<br> - Failures attributed to logo extraction phase<br> - Specific reasons for inaccuracies: <br> - Text-based logos<br> - Complex image-based layouts |

## VI. CONCLUSION

Phishing attacks remain a persistent threat due to their costeffectiveness and ease of deployment. While various prevention approaches have been employed, they struggle to eep pace with the constant evolution of phishing websites. In this study, we conducted a comprehensive review of phishing attacks and prevention methods and introduced a novel approach utilizing Optical Character Recognition (OCR) to detect phishing websites. Unlike prior research, our approach successfully addresses limitations in existing methods, offering dynamic detection capabilities and mitigating privacy concerns associated with WHOIS data in machine learning. Even in cases where the phishing server has been compromised, our method can still identify the threat. While there are areas for improvement, the technique demonstrates a high level of detection accuracy, with promising evaluation results. Our future work will focus on implementing this approach on mobile platforms, where we anticipate facing additional challenges. These include addressing the potential impact of frequent detection on the limited resources of mobile platforms, conserving power and network bandwidth, and exploring methods to enable phishing detection with minimal or no network data. We will also investigate ways to seamlessly integrate this functionality into existing mobile browsers and offload resource-intensive tasks to server-side operations to reduce the burden on mobile users. Our goal is to develop a more tailored solution to tackle the unique security challenges of mobile platforms

## REFERENCES

I. V. Wu, R. Manmatha and E. M. Riseman, “ Finding Text in Images”, In Proc. of Second ACM International Conference on Digital Libraries, Philadelphia, PA, pp. 23-26, 1997.

II. C. Garcia and X. Apostolidis, “Text Detection and Segmentation in Complex Color Images”, In Proc. of International Conference on Acoustics, Speech and Signal Processing (ICASSP2000), Istanbul, Vol. 4, pp. 2326- 2330, 2000

III. Patel C, Patel A, Patel D. Optical character recognition by open source OCR tool tesseract: A case study. International Journal of Computer Applications. 2012 Jan 1;55(10)

IV. Y. Zhang, J. Hong, and L. Cranor, CANTINA: A Content-Based Approach to Detecting Phishing Web Sites

V. Wandera, Mobile Phishing Report 2018, 2017.[Online]. Available: http: //go.wandera.com/rs/988-EGM040/images/Phishing%20%282%29.pdf

VI. Kaspersky, Financial phishing accounts for over 50% of all phishing attacks fo..., 2018. [Online]. Available: https://www.finextra.com/pressarticle/72837/financ ial-phishingaccounts-for-over-50-of-all-phishingattacks-for-the-first-time. [Accessed: 17-Nov2018].

VII. T. Vidas, E. Owusu, S. Wang, C. Zeng, L. F. Cranor, and N. Christin, QRishing: The Susceptibility of Smartphone Users to QR Code Phishing Attacks, Springer, Berlin, Heidelberg, 2013, pp. 5269

VIII. S. Chugh, Why Google is Forcing You To Have SSL Certificate on Your Websites, 2018. [Online]. Available: https://serverguy.com/security/ googleforcing-ssl-certificate-websites/. [Accessed: 17- Nov-2018].

IX. K. Krombholz, P. Frhwirt, P. Kieseberg, I. Kapsalis, M. Huber, and E. Weippl, QR Code Security: A Survey of Attacks and Challenges for Usable Security, Springer, Cham, 2014, pp. 7990.