Block Chain and Qr Code Based Malicious Transaction Analysis and Detecting the Vulnerability

[1]Medisetti Ramesh, [2]G Aruna Rekha
[1,2]Dept. of Computer Science & Engineering, KIET, Kakinada, AP, India

Abstract
In today's digital age, information and communication technology have revolutionized the way we live our lives. With the widespread adoption of the internet, consumers are increasingly relying on online services, including financial transactions. However, this has also made them vulnerable to cyber threats such as malware and web phishing, which can compromise their sensitive data.
Traditional methods of authentication and security such as usernames and passwords are no longer enough to protect against these threats. Therefore, there is a need for more robust security mechanisms to prevent attacks and safeguard personal information.
One potential solution is the use of secure QR codes as an anti-phishing tool. QR codes are versatile, easy to use, and can store a vast amount of data. By incorporating machine learning algorithms and convolutional neural networks (CNNs) into the QR code scanning process, it is possible to identify and block malicious codes that can compromise personal data.
Overall, the use of secure QR codes can help improve the security of online transactions and prevent web phishing attacks. As technology continues to advance, it is crucial to stay ahead of potential threats and implement innovative solutions to protect against cyber threats.

I. Introduction

Over the last few years, Quick Response (QR) codes have gained immense popularity due to their ability to encode a large amount of data and the flexibility to store various types of information. However, with this increased popularity comes a significant security risk, as malicious links can be embedded within the codes, leading to phishing websites and other harmful consequences. This threat is particularly concerning given that QR codes can be easily printed on stickers and placed over legitimate codes, tricking unsuspecting users into clicking on dangerous links.
Despite the widespread coverage of QR code-based attacks, research in this field has been limited, and there has been little focus on the intersection of security and human-computer interface. In order to address this gap, our paper outlines the key requirements for QR codes, per-user applications, and usability in order to advance research into making QR code processing both secure and user-friendly.
One significant challenge is that many QR code scanning apps currently on the market do not verify the validity of a URL before opening the link, leaving users vulnerable to malicious links. To combat this issue, we propose the use of secure QR code scanning tools that can verify the URL before opening the link.
Furthermore, we have developed a QR code scanning tool that is both easy to use and quick to process. This tool is intended to replace older systems that catered solely to mobile phone users, making it much simpler and quicker to refill prepaid accounts. Additionally, we recommend using QR codes to scan for malicious links in order to prevent potential security breaches.
Our research underscores the need for secure and user-friendly QR code processing tools, and provides design requirements for the QR code itself, the reader program, and usability challenges. By addressing these issues, we can make QR code processing both secure and practical, further advancing the potential and usefulness of this technology.

II. Related Work

In this paper [1], a watermarking algorithm of color image is proposed based on Discrete Wavelet Transform, Discrete Cosine Transform and Singular Value Decomposition (DWT-DCT-SVD). First convert host color image from RGB color space to YUV color space. Then a layer of discrete wavelet transform is applied to the luminance component Y, and divided the low frequency and into blocks by using discrete cosine transform, and conducted SVD with every block. Finally embed watermark to the cover image. In this paper [2], a new digital watermarking model is proposed for the medical images. An improved SMQT is used for image enhancement and the image is being segmented using OTSU thresholding. Discrete Wavelet Transform (DWT) and Inverse DWT are used to embed and extract the watermark on the host image. The goal of our scheme is to make the watermarking more robust against attacks and secure the image from privacy threats. This paper [3] presents a Wavelet transform–Singular Value Decomposition based robust zero watermarking technique for medical images to address the privacy and security issues. Unlike conventional watermarking, the proposed method conserves the reliability of the cover image without bringing any artifacts and without any change in the critical information contained in the medical image. The performance of the scheme is assessed with teleophthalmological images. The simulation results reveal the robustness of the proposed technique against various image processing attacks and indicate its suitability for safe exchange of medical images among remote medical practitioners. This research [4] is done to find the best digital watermarking technique to highly secure digital image form the illegal copies. The research work also done to analyze the possibilities of dual watermarking. Various standard research articles were studied and it is found that dual watermarking is possible with some situation. This research work motivates and offers different combinations on digital watermarking techniques in near future for efficient output of watermarking. The paper [5] proves that the contrast of XVCS is 2((k-1)) times greater than OVCS. The monotone property of OR operation degrades the visual quality of reconstructed image for OR-based VCS (OVCS). Accordingly, XOR-based VCS (XVCS), which uses XOR operation for decoding, was proposed to enhance the contrast. Advantages are: Easily decode the secret image by stacking operation. XVCS has better reconstructed image than OVCS. Disadvantages are: Proposed algorithm is more complicated. In [6] paper, present a blind, key based watermarking technique, which embeds a transformed binary form of the watermark data into the DWT domain of the cover image and uses a unique image code for the detection of image distortion. The QR code is embedded into the attack resistant HH component of 1stlevel DWT domain of the cover image and to detect malicious interference by an attacker. Advantages are: More information

representation per bit change combined with error correction capabilities. Increases the usability of the watermark data and maintains robustness against visually invariant data removal attacks. Disadvantages are: Limited to a LSB bit in the spatial domain of the image intensity values. Since the spatial domain is more susceptible to attacks this cannot be used. In [7] paper, design a secret QR sharing approach to protect the private QR data with a secure and reliable distributed system. The proposed approach differs from related QR code schemes in that it uses the QR characteristics to achieve secret sharing and can resist the print-and-scan operation. Advantages are: Reduces the security risk of the secret. Approach is feasible. It provides content readability, cheater detectability, and an adjustable secret payload of the QR barcode. Disadvantages are: Need to improve the security of the QR barcode. QR technique requires reducing the modifications. The two-level QR code (2LQR), has two public and private storage levels and can be used for document authentication [8]. The public level is the same as the standard QR code storage level; therefore it is readable by any classical QR code application. The private level is constructed by replacing the black modules by specific textured patterns. It consists of information encoded using qr code with an error correction capacity. Advantages are: It increases the storage capacity of the classical QR code. The textured patterns used in 2LQR sensitivity to the P&S process. Disadvantages are: Need to improve the pattern recognition method. Need to increase the storage capacity of 2LQR by replacing the white modules with textured patterns.

III. QR Code Attacks

The increasing popularity of QR codes as a mobile media element can make QR codes attractive targets for malware authors. However, attacks relying on QR codes are relatively new [3], [12]. We define a QR code-based attack as an attack that attempts to lure victims into scanning a QR code that directs them to malicious websites. The key idea behind QR code attacks is that victims might trust the web page or the printed material on which the QR code is displayed, and assume that the associated code is harmless. Typically, a user scanning a malicious QR code is directed to an exploit or to a phishing site. In the rest of this section, we discuss in more detail a set of realistic attack scenarios. A. QR Codes Leading to Phishing Sites Phishing attacks have received a considerable amount of attention because they are relatively simple and efficient. A phishing attack relies on both technical deception and social engineering techniques. The attacker must persuade the user to perform a series of actions that provide access to confidential information. The attack relies on the fact that a large number of users judge a website's legitimacy by its look and feel, which can be easily copied by an attacker [13]. Therefore, a phishing attack often starts by impersonating a popular website to abuse user trust. Because of the properties of QR codes (e.g., easy generation, distribution, and opacity), their adoption can increase the user's vulnerability to phishing attacks for three main reasons. First, since with QR codes URLs do not need to be manually entered anymore, users might not pay attention to the addresses they are directed to. As shown by Onarlioglu et al. [14], in a normal situation users might be able to distinguish a benign URL from its misspelled counterpart. Unfortunately, the scenario is totally different when the user is directed to a specific website via a QR code. To make things worse, mobile operating systems typically allow websites to hide their URL once the page is loaded. This is intended to improve usability on small screens, but this feature can also be used to deceive users redirected to a phishing website. Second, because of limited screen size, mobile browsers cannot display very long URLs. Therefore, a phisher can construct a long URL that starts with a legitimate name as part of the URL, but actually points to a different domain. Here again the effectiveness of the attack is improved because, thanks to the QR code, the user might never see the complete URL. Third, miscreants can use a combination of QR codes and URL shortening services to hide the malicious URL and avoid

IV. Methodology

Predictive maintenance is a crucial aspect of any modern-day business that relies heavily on machinery and equipment. The process of implementing predictive maintenance begins with collecting data from various sources. In this case, QR code links are collected for testing and training data.

Data pre-processing is the next step, where the collected data is cleaned and transformed to remove noise, outliers, and missing values. Feature engineering is also carried out to prepare the data for analysis.

Data analysis is the core of predictive maintenance, where machine learning algorithms such as regression, decision tree, random forest, and neural network are used to analyze the pre-processed data. The analysis helps in predicting when maintenance is required, identifying potential faults, and determining the remaining useful life of components.

Model evaluation is done to evaluate the effectiveness of the models developed using machine learning algorithms. Metrics such as accuracy, precision, recall, and F1-score are used to evaluate the models and identify areas for improvement.

Implementation is the next step, where the predictive models developed are implemented into the system. The implementation can involve scanning QR codes and checking if they are malicious.

Continuous improvement is an essential aspect of predictive maintenance, where the models need to be continuously updated and improved to adapt to changing conditions. The process involves monitoring the performance of the models, collecting feedback from the maintenance team, and updating the models based on new data. suspicion. As a result, unsophisticated users may be tricked by attackers [15] to visit a web page even after observing the corresponding short URL. B. Malicious Software Distribution Attackers often use malicious websites to distribute malicious software and perform drive-by download attacks [16]. In a recent drive-by download attack aimed at Android smartphones [17], malicious links were posted on online social networks to redirect victims to a malicious page. This page was designed to infect Android smartphones with malware (a variant of Android OfFake). This malware then connected to a Command-and-Control (C&C) server to join a botnet, allowing the attacker to execute arbitrary commands and exfiltrate personal information from the phone. Although there has been no report of QR codes used in drive-by download attacks in the wild, the adoption of QR codes together with drive-by download attacks is a growing concern [5]. For example, attackers might deceive victims into scanning a malicious QR code leading to a previously compromised website hosting an exploit kit. Attackers use a variety of techniques in websites serving malicious code. Such code is often hosted on legitimate websites, in a hidden HTML iframe, or in obfuscated JavaScript code leading the victim to the server hosting the exploit kit. This redirection is often done in two steps: first, the exploit kit first fingerprints the victim's device and then, based on the information retrieved, serves a relevant exploit to the device.

Figure 2 presents a simple scenario of a QR code attack in which the victim is redirected to an exploit site after scanning the QR code and visiting the compromised website. The appropriate malicious code is automatically loaded by fingerprinting the OS version and identifying vulnerable applications on the victim's device. In order to launch successful attacks, attackers often target vulnerable client applications such as web browsers, Adobe Reader, and Adobe Flash, exploiting them using common memory exploitation techniques. For example, Flash is being actively exploited in the wild in earlier versions of the Android platform by attackers that embed malicious SWF objects in HTML code.

IV. Proposed Methodology

Our aim is to analyze the prevalence of malicious QR codes on the web. For this purpose, we built a tool that: a) crawls the web and extracts image files; b) searches for QR codes in the extracted images and extracts from them any URLs discovered; and, c) identifies malicious URLs obtained from the QR codes. A. System Architecture In the following, we describe in more detail the architecture of our image crawler and the mechanism we used for detecting malicious QR codes. 1) Image Crawler Engine: The crawler engine relies on the scrapy framework [18] to extract URLs within each web page visited and append them to the crawler queue. Although we only crawl publicly viewable contents, our crawler processes robot.txt files and therefore complies with the Robots Exclusion protocol [19]. When a URL is disallowed, it is removed from the queue; otherwise, the URL is passed to the image crawler. The image crawler parses each page and extracts images using XPath (e.g., //img/@src). For each image we keep some metadata, including the download path, the primary referring URL, and the image checksum. The URL of a scraped image is used to schedule downloading of the image. Successfully extracted images are then stored in a MongoDB database along with their metadata. 2) QR Code Extractor: We use an open source QR code decoder [20], which provides a Python interface to a Java QR decoding library. While there are several QR code decoding libraries available, we chose it over other implementations because it was faster and allowed our system to scale to a larger number of images. The QR code extractor retrieves images from the database and attempts to find the three finder patterns in each image. If the special sequence of black and white pixels patterns are not found within an image, it presumes that the retrieved image is not a QR code and retrieves the next image to process from the MongoDB server. QR code metadata, including the URL of the website from which the QR code was extracted and any extracted target URL, is inserted into the image collection. 3) URL Matcher: URL matching is the final phase of detecting malicious QR codes. It compares the URLs extracted from the QR codes to a list of malicious websites created from a number of publicly available resources. To populate this list, we deploy a crawler to collect domain feeds provided by major URL blacklists (PhishTank [21], Malware Domains [22], the Malware Domain List [23], malc0de [24], Malware Block List [25], and vxvault.siri-urz [26]). Our system collects on average 1,600 entries per day from these sources. From these feeds, we generated a collection of approximately 640,000 unique malicious URLs to perform URL matching. The URLs obtained from QR codes are then compared with this list to identify malicious QR codes and the type of malicious actions for which they were intended. 4) Web Crawling: A suitable selection of seeds, or starting points for the web crawler, is an important precondition for an efficient and high-coverage crawler. We therefore selected a combination of sites from various sources in order to extract images from different categories of sites and increase the chances of finding possible attacks. In particular, we selected the initial crawling seeds from six different categories, including free downloads, online games, adult, music, online news, and personal/business websites. For each category, we extracted the results of English language Google searches. We also included the top 1,500 most popular websites as published by Alexa [10]. In addition, we added to the list some websites that were more likely to contain malicious QR codes.

V. Modeling and Analysis

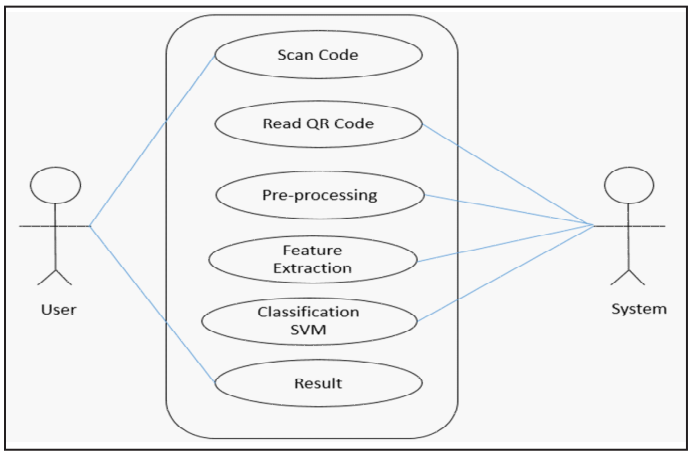

Fig. 1: Model Building diagram

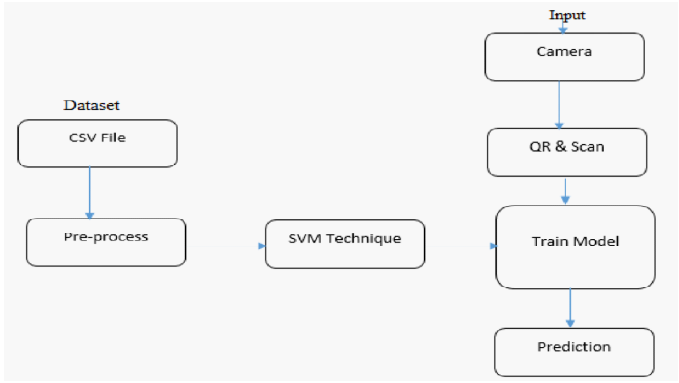

Fig. 2: System Architecture

VI. Proposed Algorithm

The proposed system is designed to provide a robust and reliable solution for data processing and analysis. The web-based application employs a Python-built backend and a PHP and CSS front end to create a user-friendly and interactive platform. The connection between the backend and frontend utilizes the MySQL database for data storage and retrieval. To ensure the accuracy and effectiveness of the predictive model, data is collected from numerous sources across the nation. The collected data is then split into two parts, with 80% of the data allocated for training and 20% for testing.

To prepare the data for analysis, unwanted data or null values are removed from the dataset during the preprocessing stage. The data is then subjected to different machine learning approaches, such as support vector machines, to recover the features required for analysis.

After recovering the features, the model is trained on the training dataset, and the accuracy and effectiveness of the model are evaluated using various metrics. Once the model is deemed satisfactory, features from the input data are compared using it

to make accurate predictions.
Overall, the proposed system is a powerful tool for data processing and analysis, with the ability to provide accurate and reliable predictions. With its user-friendly interface and robust architecture, the system is wellsuited to a wide range of applications, from predictive maintenance to fraud detection and more.

VII. Conclusion

The prevention of malware and phishing attacks using QR codes has been discussed. We started by examining the current status of QR code scanners' capacity to recognize fraudulent urls. Later, we presented our method for more accurately detecting malware urls using Support Vector Machine techniques. The provision of scalability, flexibility, and security for secure communication is our main objective when developing technology.

References

[1] M. Rajeswari, M. Revathi, Mr. J. Marimuthu, "IRIS Recognition using QR Code for Finding Duplicate Certificate", IJERT 2020
[2] S.Hariswetha, S.Indira, S.Latha, T.Sivabharathi,"QR based Automatic Penalty Charging for Violation of Traffic Rules" (IJERT)2020.
[3] SarojGoyal, Dr. Vinod Kumar, Dr. SurendraYadav, Manish Mathuria," Quick Response Code Implementation in Society", IJERT,2019.
[4] Abhishek Mehta, Dr. KaminiSolanki,"Design and Development of QR Code Recognition from Digital Image",(IJERT)2019.
[5] K. Balasubramanian, P. Suhashini, V. Priyanga, K. Kavinila, "Using QR Code Detection and Recognition Total Variations and LDA Approach",(IJERT) 2018.
[6] N Rodimawati , I G P A Budijahjanto, R E Putra and AY Wicaksono , "A Responsive Web-Based QR Code for Inventory in The Laboratory of Informatics, UNESA", The 2nd Annual Applied Science and Engineering Conference (AASEC),U NESA I0P Conf. Series: Materials Science and Engineering 288, (IJERT) 2018.
[7] Milind Amrurkar Dr.Anup Palsokar Asst.Prof. Pankaj Raibagkar, "QR Code based Stock Management System ", The Journal: International Research journal of Engineering and Technology (IRJET) e-ISSN: 23951056 p4SSN: 23910375, (IJERT) 2017.
[8] G.M. Agusta, K. Hulliyah, R. B. Bahaweres, et al. QR code augmented reality tracking with merging on conventional marker based backpropagation neural network. In Advanced Computer Science and Information Systems (ICACSIS), 2012 International Conference on, pages 245–248. IEEE, 2012.