Journal of Cyber Security

Tech Science Press

Doi:10.32604/jcs.2025.059398

REVIEW

Quick Response Code Security Attacks and Countermeasures: A Systematic Literature Review

David Njuguna* **and John Ndia**

School of Computing and Information Technology, Murang'a University of Technology, Murang'a, 75-10200, Kenya

*Corresponding Author: David Njuguna. Email: dnganga175@gmail.com

Received: 07 October 2024; Accepted: 21 January 2025; Published: 18 February 2025

ABSTRACT: A quick response code is a barcode that allows users to instantly access information via a digital device. Quick response codes store data as pixels in a square-shaped grid. QR codes are prone to cyber-attacks. This assault exploits human vulnerabilities, as users can scarcely discern what is concealed in the quick response code prior to usage. The aim of the study was to investigate Quick Response code attack types and the detection techniques. To achieve the objective, 50 relevant studies published between the year 2010 and 2024 were identified. The articles were obtained from the Institute of Electrical and Electronics Engineers, Elsevier, Springer, Science Direct, Wiley, Association of Computing Machinery, and Google Scholar. From the study, Quick Response-Quick Response attacks, Quick Response code payment attacks, Quick Response code counterfeiting, and QR code information leakage have been identified as potential Quick Response code security threats. Barcodes can be maliciously used to run different attacks such as phishing, pharming, malware propagation, cross-site scripting, and Structured Query Language/command injection and reader applications attacks. To mitigate against Quick Response code attacks, various techniques such as cryptographic schemes, machine learning, artificial intelligence, two-factor authentication, One-time password, and mutual authentication schemes have been used. Users must remain vigilant when scanning Quick Response codes and take steps to verify their legitimacy. More research is needed to develop automated detection techniques that can authenticate QR codes and detect malicious URLs or malware in real time.

KEYWORDS: QR code; detection; barcode; QRishing; cryptography

1 Introduction

A barcode is a graphical representation of information through distinct patterns. In a one-dimensional (1D) linear barcode, data is encoded as a series of vertical lines with varying spaces, while a two-dimensional (2D) barcode combines vertical and horizontal squares. This encoded data can be retrieved using imaging devices such as barcode scanners or smartphones with specific reader apps. A QR code is a type of 2D barcode [1]. Various types of data, including binary, alphanumeric, numeric, and Kanji characters [2], can be stored in this format. As the QR code version increases, the data capacity also expands.

DensoWave created the QR code to control production at its industrial facilities [3]. Since then, its use across several industries has increased dramatically. It is employed as a follow-up digital action, such as visiting a website or logging in, for marketing purposes, as an extra information connection, for security-related tasks like device pairing or authentication, and to connect digital and physical locations [4]. It serves as a shortcut [5] for quick access to digital information on the internet. QR codes may encode music, pictures, Uniform resources Locator (URLs), etc.

The smartphones' ability to read and instantly access the URL encoded in them, prompts advertisers to utilize QR codes to promote their online presence. They are frequently seen on public signs, usually pointing people to a website that offers further details on a certain brand, business, or area. A recent survey found that 75% of shops currently provide their consumers with this 2D technology so they may interact with and follow potential customers [2]. Consumers can obtain coupons, discounts, or other product or service-related information by scanning them with QR code scanners on their smartphones or tablets. However, users must scan a QR code to decipher its contents because their encoding strategy is more machine-readable than human-readable. Because of this, attackers can easily conceal and initiate their attacks using QR codes that contain harmful material, such as phishing URLs [6]. In addition, people frequently give in to their curiosity and scan the codes they see. QR codes are among the largest hidden security risks because of their inherent obfuscation and curiosity [7]. Human curiosity is a serious weakness in QR code security when hackers take advantage of people's inclination to investigate novel or fascinating prospects, particularly when they seem innocuous or alluring. Because QR codes are straightforward, unobtrusive, and simple to use, they can easily trick users into taking activities that jeopardize their security.

Data can be extracted from a barcode image using a barcode scanner, an optical device with imaging and processing capabilities [8]. The popularity of smartphones with high-resolution cameras has encouraged programmers to produce mobile apps that can decode barcode pictures and offer extra functionality like messaging, exchanging contacts, and URLs. The most often used barcode types are Quick Response codes since they have the most data capacity. Given how quickly everyone and every industry has adapted to it, it is unsurprising that phishers use QR codes' vulnerability to start their attacks. Phishing is an attempt to steal sensitive information from the target, such as bank account details or login credentials, using either technical or social engineering techniques [9]. There will always be a danger or weakness that could compromise user security and privacy, just like with every new technology [5].

Numerous countermeasures from software-based to user-centered, have been suggested for deterring QR code attacks. User-centered countermeasures include programs that, increase user awareness by teaching people about the potential uses of QR codes in phishing and pharming attacks. Additionally, users might be informed and reminded to never submit sensitive information using a QR code link and to always verify the URL that the code points to [10].

This study aims to investigate Quick response codes security vulnerabilities and countermeasures. To achieve the objective, the study addresses the following research questions:

RQ1: What are the different types of QR code attacks?

RQ2: What techniques exist for detecting QR code attacks?

The other sections of the paper are structured as follows: section two focuses on the types of QR code attacks; Section three looks at the techniques for detecting QR code attacks; Section four presents the methodology, findings, and discussion and finally conclusion in Section five.

2 Types of Attacks on QR Codes

Generally, QR codes have various sections set aside for distinct uses. There are portions of the QR code that are functioning and cannot be retrieved via error correction. Modules of black and white are used to encode the data [8]. Scanners are still vulnerable to QR code attacks [8,11,12]. QR codes can be fully replicated over or only slightly altered when utilized as an attack vector. If singular modules are inverted from white to black or *vice versa*, we consider partial alterations. Generally, we separate two categories of harmful alterations. The first method modifies both black and white pixels, while the second method just modifies changes from white to black. The latter case is similar to what happens when an attacker uses a black pen to

alter an already-existing QR code, as details [13]. Given a QR code's structure, changing one or more of the modules can have an impact on the code: The data is interpreted differently by the QR code scanner when the character encoding is changed. The length of the data is indicated by the character count indication, and tampering with it can cause the scanner to read less data or interpret modules that aren't meant to hold data. Moreover, other encodings can be combined with the data to alter the modes or add and remove certain portions [12].

Control character abuse may allow hackers to hide malware [8]. Attacks carried out through QR codes mostly target automated procedures and human communication. Cross-site scripting attacks and browser-based exploits can both be made improper use of QR codes [14]. Potential challenges to QR code security include QR-in-QR attacks, QR code payment attacks, QR code counterfeiting, and QR code information leakage [15].

Studies show that a variety of malicious attacks, including phishing, virus propagation, cross-site scripting (XSS), SQL/command injection, and reader application assaults, can be carried out maliciously via barcodes [8].

According to a study by [16], there are new automated two-layer QR code attacks that can encode two different messages in a QR code that can be independently read by reorienting the scanner. Through the use of a QR code generator, research by [17] presents QR code attack scenarios for Bitcoin payments. According to a different study [18], a nested QR code combines two QR codes simultaneously in a square area so that, depending on the orientation of the scanner, both are easily readable.

2.1 QR-in QR Attacks

Attackers generate a malicious QR code and digitally insert it into the image of a legitimate QR code or overlay it in physical form (e.g., a sticker on a printed QR code). Users think they are scanning the original QR code but unknowingly interact with the malicious one. When scanned by a device, the malicious QR code overrides or coexists with the original, directing the user to the attacker's payload [19]. Some scanners or apps may process the malicious QR code instead of the legitimate one, depending on the scanning method. The embedded malicious code may redirect users to phishing sites, download malware, etc.

2.2 QR Code Phishing Attack

With the expanded utilization of QR code technology, QR code phishing, or QRishing offers a new line threat to this technology [20]. In a barcode phishing assault, the attacker tries to obtain sensitive data, such as the user's credit card number and login credentials, by, for example, embedding a malicious Web address inside the barcode that takes the user to a phony website that mimics the real one quite closely [14,20,21].

Users can scan QR codes to open websites without having to type in the URL. Additionally, some mobile browsers hide the URL to increase usability. Even when the URL is displayed, users may not be able to see it due to screen size constraints on smartphones, which makes QR codes an extremely appealing vector for phishers. Furthermore, some attackers conceal the genuine URL using URL shortening techniques to deceive unsuspecting users [20]. Phishing for cryptocurrency can be as simple as sending out unsolicited emails purporting to be from a specific website. In this instance, letters are delivered on behalf of Bitcoin exchanges or wallet websites. These spoof emails appear notably more organized, skillfully written, and thorough than typical phishing texts. It can be a security alert with an embedded QR code that states. On the wallet website, the user may have set up these settings themselves to receive these notifications; in this case, they won't see anything amiss. In one case, the victim will be prompted to enter their wallet information on a phony Bitcoin service website. The most well-known Bitcoin web wallet's website appears quite straightforward but

is instantly recognizable, which makes it easier for thieves to successfully counterfeit it. Another possibility is to generate harmful QR codes designed to exploit vulnerabilities in phones and other scanning devices [22].

2.3 Malware Propagation

According to research by [23], attackers can utilize QR codes to reroute consumers to malicious websites that employ the device's vulnerable software to surreptitiously install malware. Usually, an exploit kit is used for this, which uses the device's fingerprint to identify the right virus and exploit it. An attacker may embed a URL in a QR code such that, upon code reading, the victim is taken to a page where malicious software is downloaded automatically. In this manner, an attacker can introduce viruses, spyware, Trojan horses, or worms into a system, seriously harming both the user and the machine. Research indicates that Kaspersky Labs found multiple malicious websites with QR codes for mobile apps, one of which had a Trojan that could text premium-rate short numbers [15].

An Android-based malware that employs QR codes as an attack vector was detailed by McAfee Labs. Although the malware's payload and code are extremely similar to those of other well-known examples, this variant is different in that it spreads using a straightforward QR code. When the trojanized program is installed, the code starts to download and delivers Short Message Service (SMS) messages to premium lines that bill consumers for enormous amounts of money [24]. An example of a malware attack is where attackers send SMS to premium rate numbers by using a QR code that contains a malicious link, which directs users to a malicious website where they unintentionally download Jimm, which is infected with TrojanSMS.AndroidOS.Jfake.f malware [15].

2.4 Barcode Tampering and Counterfeiting

Given that QR codes can be used to give product information, an attacker may profit by pasting a phony QR code to promote bogus products or special deals, with the intention of convincing victims to purchase another product [25]. Barcode tampering could make it possible for an attacker to counterfeit a product when barcodes are used to identify and track objects, like in a supply chain.

2.5 Barcode-in-Barcode Attacks

A unique instance of tampering occurs when a pattern in a picture of a barcode matches another legitimate barcode [25]. The secondary barcode may occasionally be decoded, leading to potentially hazardous circumstances, however, most of the time the QR code will scan normally. This has to do with two separate software decoding a file type differently. Remarkably, we found that barcode-in-barcode decoding could occur by accident, particularly when barcodes are dense [26].

2.6 SQL and Command Injections

SQL injection attacks occur where the hacker adds an extra SQL statement or command to the end of the pre-defined inquiry statement, tricking the database server into carrying out an illegal activity to steal important data or take over the database server [27]. In automated systems, barcode-encoded data can be stored and accessed via databases using QR codes. An attacker may quickly launch a SQL injection attack if the string contained in the barcode is attached to the query without first being properly sanitized [21].

2.7 Cross-Site Scripting Attacks (XSS)

In XSS attacks, an attacker can run malicious scripts on the victim's web browser, leading to a multitude of negative outcomes, including data compromise, cookies, passwords, and credit card number theft, among

other things [28]. Considering that mobile apps frequently rely on Web technology, malicious JavaScript code can be inserted into legitimate HTML pages and run within the app. This might happen, for instance, if the server fails to sanitize user data before it is rendered on a page [29].

2.8 Reader Applications Attacks

Many barcode reader apps request full permissions to access user resources, such as images, contact lists, and device locations, during installation. If a vulnerability exists that can be triggered by a carefully crafted barcode, it could allow attackers to access private user data [13].

3 Techniques for Detecting QR Code Attacks

Traditional detection methods are ineffective due to the increase in hacking capabilities and the diversity of attack types brought about by the rapid evolution of technology [30]. Numerous methods for detecting QR codes have been developed by researchers.

To mitigate the security risk associated with barcodes that hold private or sensitive data, the authors [31] suggested a barcode-based secret-sharing method. The suggested method creates a secure data transfer plan using a QR code and a secret sharing mechanism. The secret sharing scheme's basic concept divides a secret into n shadows, also known as shares. The original secret cannot be decoded by anyone using their share. Only when any t out of n shadows (t <= n) are held together can the secret be retrieved. Shamir's secret sharing system serves as the foundation for the proposed plan. By using a secret sharing approach, the secret data is split up into shares of shadows. Every QR-code tag has an embedded version of the created shadows. If the predetermined threshold for received shadows is not met, it will be impossible for anyone to read content directly from QR codes.

Two-dimensional quick response (QR) codes can be deceptive since it might be challenging to distinguish a malicious QR code from an authentic one. The Quick Response Code Secure, or QRCS, was proposed by the authors [20]. Using digital signatures, QRCS is a universally applicable, efficient, and effective solution that solely focuses on the originator's validity and, by extension, the integrity of the QR code. Utilizing digital signatures and hash functions, the QRSC is a client-server-based cryptographic system that ensures the integrity and validity of QR codes. Every scanned QR code is verified using its digital signature and public key. When the verification process fails, the QR code is deemed malicious and is blocked or obstructed at the initial scanning stage.

A safe artificial intelligence barcode scanner was proposed by the authors [19] to identify harmful linkages included in both 1D (linear) and 2D QR codes. A barcode counterfeiting-based cybercrime assault that can be used for online attacks has been presented. Several resources were used to construct a dataset of 100,000 benign and malicious URLs, from which lexical characteristics were extracted. To show how various elements and users interact with online barcode material, analyses were carried out. Numerous versions of artificial intelligence were put into practice. The best model for recognizing fraudulent URLs was found to be a decision tree classifier with an accuracy of 90.243%

The authors [7] proposed SafeQR, a QR code solution that improves the detection rate of malicious URLs by utilizing two security APIs already in place to detect malware and phishing attacks: Google Safe Browsing API and PhishTank API. This was done in response to the existing QR code scanners' poor performance. To help consumers better notice warnings, a meticulously developed and implemented visual warning scheme was also included. The authors designed and carried out user research to evaluate the scheme's efficacy with the techniques used by other QR code readers already on the market. Using 2D barcodes, online spammers can inundate the physical side of the Internet, deceive users into viewing or accessing unwelcome and

irrelevant content, and potentially undermine the credibility of legitimate content. This problem of IOT web spamming must be solved.

To solve the issue of spamming the Internet of Things, the authors [32] suggested using the Elliptic Curve Digital Signature Algorithm (ECDSA). A digital signature embedded inside a 2D barcode helps to verify the authenticity of the content and ensures its integrity. It is necessary to have a QR Code generator that can encode content into a QR code and digitally sign it. By using public-key cryptography, the content, a public-private key pair, and certificate information, it creates a digitally signed QR code and the necessary certificates. Three pieces of information are encoded in the modified QR code: the content creator's public key (PK), digitally signed (DS)material, and the original content/message (C). Since the content refers to a website, the certificates confirming the author's identity are positioned at the URL. The second component is a mobile application that can verify the certificate chain and verify the integrity of the content contained in QR codes by scanning them.

The BarSec desktop application and the BarSec Droid Android were created by the authors in [8]. These applications utilize the ZXing library and symmetric and asymmetric cryptography techniques to create and scan safe and functional barcodes. The JSON structure is used by the barcode generator, which is exclusive to the BarSec desktop application. It provides several security features, such as data integrity, confidentiality, access control, and barcode authentication. It may also be used to create and read QR codes, and also offers usability warnings. The ECDSA and Rivest-Shamir-Adleman (RSA) are two separate digital signature algorithms that are supported by both the BarSec desktop application and the BarSec Droid Android application. Both applications employ the Secure Hash Algorithm (SHA-256) as their hash function. They allow various key lengths for the hash-based message authentication code (HMAC). Furthermore, they employ four modes of the Advanced Encryption Standard (AES): Galois/Counter Mode (GCM), Cipher Block Chaining (CBC), Output Feedback (OFB), and Cipher Feedback (CFB). To address the challenges with usability, these techniques have undergone extensive study. Access Control Lists (ACLs), which the authors also incorporated, enable the generator to have many data layers, or numerous users with restricted access to data. It supports barcodes created by the BarSec Droid Desktop application and barcodes that have Access Control Lists (ACLs). It employs the JSON structure. It is capable of reading regular QR codes that don't adhere to any particular structure or contain encrypted data. It uses the Norton SafeWeb service to verify the web content of entire URLs that are enclosed in barcodes.

A study [33] suggested a three-tier method to handle the security concerns and vulnerabilities related to QR codes, such as the possibility of malicious code running, sensitive user data being stolen, privacy being violated, and identity theft. The integrity of the data and the QR code generator's authentication are the main concerns of the solution. Asymmetric key cryptography is used in the first tier by Digital Signature Algorithms (DSAs), which employ the SHA-1 hash function as the digital signature. For more efficient detection and prevention of dangerous actions, the verification server in the second layer uses a firewall, intrusion detection, and anti-virus software that is superior to that which is installed and used on the client's mobile phones. Through utilizing a sandbox, the third tier guards against invasions of privacy, access to personal data, and acquisition of mobile device control via the scanner application. To solve the weaknesses in QR codes, a secured QR (SQR) was developed by another study [34]. Before creating the QR code, the data is encrypted using Advanced Encryption Standard (AES) by the SQR. The password-protected data must be entered by the user for the data to be unlocked and retrieved. This method is called Secure QR code. The Fig. 1 shows the flowchart of SQR.

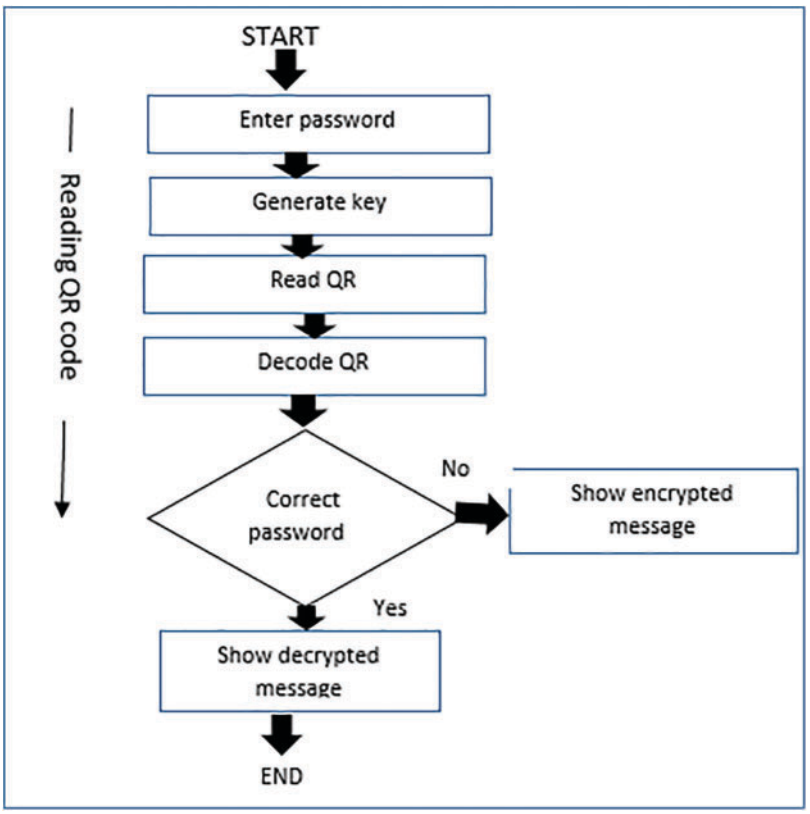

Figure 1: SQR flow chart

The authors [35] created the Quick Response Protocol (QRP), a secure two-factor authentication (2FA) method that combines a password and a smartphone with a camera to function as an authentication token. The user opens the bank website to conduct online banking transactions. Following registration, a QR Code is shown on the page, which the user can then scan with a QR Code scanner. The random number, which is produced by the random number function, and the IMEI number of a phone that the user has registered are combined to form a string that is produced by the scanning result. There is a two-factor authentication setup. The created string is automatically entered into the login page if the smartphone is connected to a network, at which point the bank's homepage opens. If not, a six-digit Personal Identification Number (PIN) is created, which must be manually entered on the login page before the bank's main website becomes accessible for transactions. The system of two-factor authentication is employed. When a smartphone is connected to a network, the created string is immediately entered into the login screen, opening the bank's homepage. On the other hand, a six-digit PIN code is produced and manually input onto the login page, after which the bank's home page becomes accessible for transactions.

2FA significantly enhances the security of QR code transactions by adding a layer of verification to ensure that the person initiating or approving a transaction is authorized to do so.

The authors [36] proposed a mutual authentication system that balances strong security with user-friendliness, specifically for quick response (QR) codes—a type of two-dimensional barcode easily scanned by smartphones. The protocol, when integrated into a QR code, can differentiate between public and private user information without compromising the image quality. The protocol is composed of three principals: the registration center, the institution agency, and the service user. Anybody who wants to use a range of

services, including an online bank, a Wi-Fi network, or cloud services, is considered a service user (SU). It is up to an institution agency (IA) to confirm if SU has access to the service provider. An exclusive function of a registration center (RC) in the system is to receive registrations from SUs and IAs and retain pertinent data that they give.

Authors [37] suggested using two-dimensional barcodes in place of security cards, which were vulnerable to hacking, in banking systems to verify user identity and remove the issue of phishing. The One Time Password (OTP) that users receive from the existing online banking system can be compromised in transit. To get around the security card's shortcomings and inconveniences. The suggested method uses a mobile OTP with a QR code to increase security and convenience.

A study [38] suggested a hybrid strategy that combines the use of digital watermarking, a data-hiding technique, One Time Password (OTP), and Quick Response code (QR code) to mitigate against security assaults in banking systems, such as phishing and pharming. The bank generates the QR code by using the OTP as the key to the watermark sequence. In this case, the initial step of authenticity verification can be completed by OTP. The sequence for the second level verification is hidden during the watermarking process by using the Hadamard transformation. During an online transaction, great security is enabled by the combination of OTP and watermark sequence. One benefit of this strategy is that the QR code can be created within a tiny space and with the proper real-time extraction method. Performance can be assessed using an Android application, which also allows for real-time extraction using a smartphone.

By adding undetectable or inconspicuous information to the QR code that may be utilized for tracking, tamper detection, and authentication, digital watermarking significantly improves QR code security. For QR codes, digital watermarks can be used as an extra layer of verification to make sure the code is coming from a reliable source.

The authors [39] suggested a countermeasure against cross-site scripting attacks (XSS) that permits restricted access to the data provided by QR codes. A solution called XSStudent was developed, which examines the retrieved URLs and contrasts them with a system that has already been trained. This investigation was conducted using a controlled attack against university users who were able to view an infected website with a JavaScript code that enabled a successful cross-site scripting assault by using a leaflet with a QR code and a fictitious link.

The authors [40] suggested employing a combination of steganography and cryptography to improve the security of the message contained in the QR Code. This combination is essential since it can provide an additional layer of protection. The suggested approach is based on a four-layer concept, where each layer provides an additional layer of protection. The asymmetric cryptography algorithm, also known as elliptic curve cryptography, is the first layer. It creates a common key and hashes it into a secret key. The Authenticated Encryption with Associated Data (AEAD) algorithm was employed by the second layer for both encryption and decryption. The cipher text and nonce are hidden inside the QR Code using the third layer, or steganography technique. To prevent a man-in-the-middle attack, the fourth layer is a QR Code that is encoded into the cover image using the One Time Pad technique and the least significant bits.

Using a malicious URL detection framework, the authors [15] suggested QsecR, a safe and private QR code scanner, to identify malicious URLs. QsecR is an Android QR code scanner that uses 39 classes of blacklist, lexical, host-based, and content-based features to classify static information in a preset manner. 4000 randomly generated real-world URLs were collected from URLhaus and PhishTank to create the dataset. Several QR code scanners assessed the security and privacy of the QsecR. In comparison to the current secure QR code scanners, QsecR performs much better in the trial, achieving a detection accuracy of

93.50% and a precision value of 93.80%. QsecR is also among the least privilege-permission applications that are the most privacy-friendly. To cope with the vulnerability to monetary loss or disclosure of personal data.

A unique QR code, mQRCode was created in a study [41] that uses patterns that display a particular spatial frequency as a kind of concealment. The original QR code appears as a Moire pattern when the targeted receiver holds a camera at a specific location. The only thing visible from any other angle is the hidden QR code.

Through the use of a multi-frame decryption technique, the decryption rate of mQRCode was greater than 98.6% in 10.2 frames throughout the testing. When cameras were placed 20° off-axis or more than 10 cm away from the intended site, the decryption rate decreased to 0%, showing that mQRCode is resistant to attacks.

A novel approach to QR Code authentication and phishing detection has been put out to distinguish between legitimate and phishing URL codes [42]. In addition to preventing the user from validating the QR Code, the suggested model will identify phishing and harmful URLs throughout the validation process. The creation of this application will aid in preventing users from falling for malicious QR Code tricks. Reading the QR code is the first step in the procedure. If the app detects the image of a QR code, it will begin to determine and investigate whether the code is authentic or fraudulent. The user will be redirected to the website if the URL is authentic. If not, the user will see a warning message from the application.

A study [43] suggested rotating the QR code at different random angles to address financial loss and privacy of personal information. One can access the QR code only if three consecutive random angle scans match the random values on the server. Until three consecutive random rotation angles match the random values on the server, an attacker attempting to access or record the dynamically rotating QR codes will not be able to access the data encoded in the code.

A study [44] suggested SafeQR codes as a countermeasure to the vulnerability of QR codes to tampering, or Quishing. These codes solve this issue by offering creative design ideas that improve QR code security. By utilizing safe design principles and visual components, the method seeks to increase the visibility of tampering, enabling users to identify and steer clear of potential phishing risks. The researchers also emphasized the shortcomings of the user-education strategies currently in use to counteract Quishing and suggested several attacker models specifically designed to deal with Quishing attacks. They also present a multifaceted protection plan that combines user awareness with innovative design. Because this approach doesn't rely on specialized knowledge or technological tools, people of different socioeconomic backgrounds won't be at a disadvantage. By putting these design improvements into practice, all users irrespective of their financial status or educational background can experience a safer online environment and have equal access to digital safety measures.

To solve the security flaws in QR codes, the authors [45] suggested QR Shield, a dual machine learning-based solution. The Random Forest (RF) and XGBoost algorithms were included in the model after several machine-learning techniques were thoroughly tested. Using a benchmark dataset of URLs, the QR Shield uses these advanced machine-learning techniques to reliably identify and detect fraudulent URLs encoded within QR codes. The QR Shield demonstrated a strong potential for identifying malicious URLs contained in QR codes, indicating that the proposed QR Shield can be broadly used in a variety of real-world domains and applications. The Anti-Malware Phishing Scanner (AMPS) is a QR code scanner that has built-in malware detection capabilities, as suggested by [46]. The AMPS scanner uses the Advanced Encryption Standard (AES) method to provide encryption and decryption capabilities for QR codes. The technique achieved an accuracy of 96.8%.

The authors [47] suggested a method for identifying phony QR codes by utilizing the QR code's error-correcting mechanism. Three approaches to error detection were presented: one based on the quantity of errors, another on the locations of errors, and a third one based on the symmetry and locality of errors. By examining the data gathered from the error-correcting procedure, the method attempts to categorize whether the QR code is authentic or fraudulent.

A method was developed by authors in [48] that would enable QR code providers to authenticate human scanners, thereby making auditing and authorization easier. The front camera of the code provider is used to concurrently confirm the scanner's hand when a phone is held close to scan a QR code. To acquire typical hand geometries, the scanner does not need to offer a stretched palm; instead, it may identify the geometry of a person's hand when it is gripping a phone. The authors created a vision-based method for obtaining biometric information from the grasping hand. This method uses the screen of the QR code to shine light onto the hand that grips the scanner, making sure that there is enough light even in low light. The user must click touchscreen buttons to begin scanning QR codes, so use MediaPipe, a hand-tracking tool, to detect and localize the hand. Next, create a transformer-based algorithm to validate four types of gripping hand biometric features extracted from the hand image: hand contour, skeleton, color, and surface. Finally, capture the subtle hand joint movements for liveness validation.

The authors [49] introduced an automated detector designed to identify QR code images and alert users to their benign or malicious nature. To develop this system, a dataset of 10,000 QR code images, including both legitimate and malicious examples, was compiled. Convolutional Neural Networks (CNNs) were employed to distinguish between harmful and authentic codes. Additionally, histogram density analysis was incorporated to enhance feature recognition and improve classification accuracy. The method utilized CNN models ResNet50, MobileNetV3, and InceptionV3 to evaluate their ability to detect malicious codes. Among these, the InceptionV3 model demonstrated the highest performance, achieving 98.13% accuracy, making it a highly effective solution for document forensics and counterfeit detection.

A study [50] proposed the Swarm-Intelligent Squirrel-Search Optimized Artificial Neural Network (SISSO-ANN) as a method for detecting malicious URLs embedded within 2D QR codes. The research highlighted the potential for cyberattacks using forged barcodes. A dataset of harmful and legitimate URLs was compiled from various sources, with their lexical features analyzed. Experimental results revealed that SISSO-ANN outperformed existing techniques, achieving a detection precision of 93.80% and an accuracy of 93.50%, surpassing the capabilities of current secure QR code scanners.

4 Methodology

4.1 Methodology

The PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) methodology was employed as a framework for this systematic literature review. This approach is recognized for its high-quality guidelines, ensuring thorough data collection and a detailed, rigorous process. PRISMA's structured procedures enhance consistency, transparency, and adherence to high standards, making it a reliable tool for producing comprehensive qualitative research reports.

To achieve the research objective, a systematic literature review (SLR) was conducted to identify various types of QR code attacks and corresponding mitigation techniques. The review involved searching databases such as Institute of Electrical and Electronics Engineers (IEEE) Xplore, Association of Computing Machine (ACM), Elsevier, Springer, ScienceDirect, Wiley, and Google Search. Table 1 provides a summary of the data sources used. The search spanned publications from 2010 to 2024, focusing on journals and conference papers. Articles not written in English or unrelated to QR code attack mitigation were excluded. The selected

articles were assessed using six criteria: source, method/technique, attack strategy, detection strategy, and others. Keywords included terms such as "QR code," "QR code security," "QR attacks," "QR code attack detection techniques," and "QR code attack prevention." The initial search yielded 70 articles, which were screened for relevance based on their titles and abstracts. These papers were then further evaluated for their eligibility in the study.

Table 1: Data sources

Data source	No. of papers
IEEE	17
ACM	6
Elsevier	4
Springer	4
Wiley	1
Google search	17
Science direct	1

4.2 Inclusion and Exclusion Criteria

A set of inclusion and exclusion criteria was applied to select relevant articles for the study. Articles were included if they met the following conditions: (1) published within the last 13 years, (2) written in English, and (3) focused on QR code security, vulnerabilities, and detection measures. Conversely, articles were excluded if they (1) were review papers, (2) did not directly address QR code security challenges, (3) were published before 2010, or (4) lacked relevance to QR code attacks. After applying these criteria, a total of 50 articles were deemed eligible and included in the final study. Table 2 presents the publications from 2010 to 2024, while Table 3 lists the number of articles excluded for not meeting the inclusion criteria. Additionally Table 4 shows a summary of SLR process.

Table 2: Summary of the studies included in the systematic review

Year of publication	No. of publications
2010	2
2012	4
2013	2
2014	8
2015	3
2017	3
2018	5
2019	8
2020	5
2021	3
2022	1
2023	2
2024	4
Total	50

Table 3: The number of articles excluded

Reason for exclusion	Number of articles
Review papers	9
Published before 2010	8
Lacked a focus on QR code attacks	3

Table 4: Summary of SLR

Criterion	Eligibility	Elimination
Identification	Journal articles, relevant internet resources Databases searched: IEEE Xplore, ACM, Elsevier, Springer, ScienceDirect, Wiley, Google Search Search terms: "QR code," "QR code security," "QR attacks," "QR code attack detection techniques," "QR code attack prevention"	Review articles Books chapters, blogs, encycropedia, articles unrelated to QR code security challenges, articles lacking relevance to QR code attacks
Language	English	Non English
Subject area	Computer science	Subjects not related to computer science discipline
Timeline	2010–2024	Below 2010

The diagram in Fig. 2 below shows a flowchart for the PRISMA methodology.

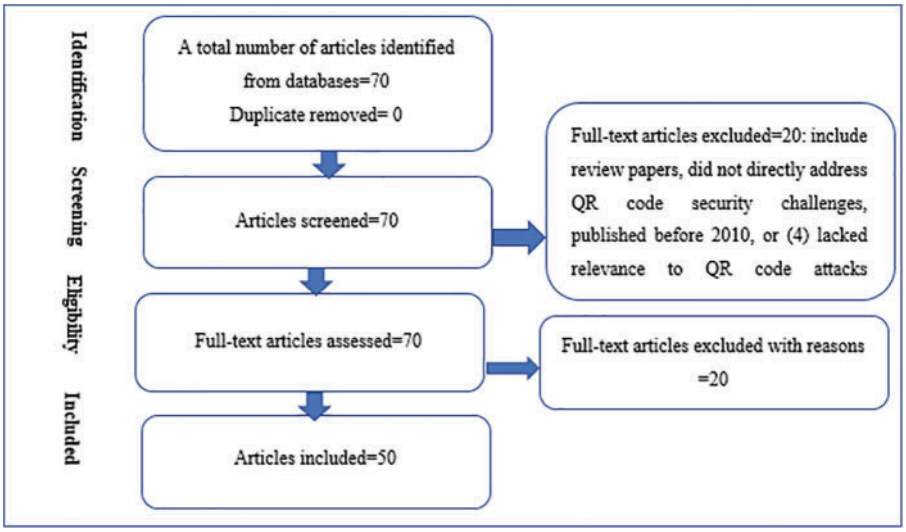

Figure 2: PRISMA methodology flowchart

4.3 Findings and Discussions

The study aimed to answer the following research questions;

4.3.1 RQ1: What are the Different Types of QR Code Attacks?

To address the first research question regarding the types of QR code attacks, a comprehensive literature review was conducted. The findings revealed various forms of QR code attacks, including QRishing, malware propagation, barcode tampering and counterfeiting, barcode-in-barcode attacks, XSS, SQL, and command injections, as illustrated in Fig. 3. Among these, malware propagation and phishing were identified as the primary security threats [15].

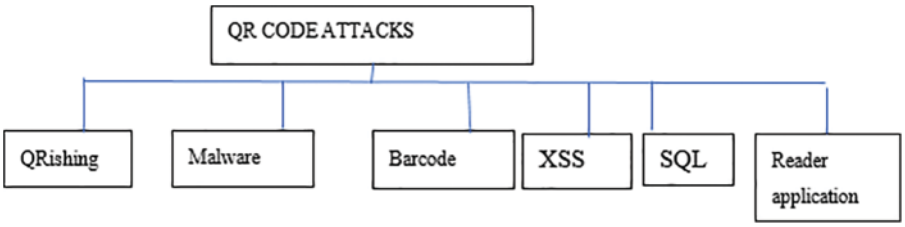

Figure 3: Types of QR code attacks

Threat actors favor QR codes due to their simplicity in transitioning users from desktops or laptops to mobile devices, which often lack robust anti-phishing protections. Additionally, smartphones, commonly used for scanning QR codes, frequently have security vulnerabilities that attackers exploit to deploy malware, steal data, or carry out other malicious activities. Attackers are also employing increasingly sophisticated methods, such as combining QR code attacks with other cyber threats like ransomware or taking advantage of browser vulnerabilities.

The ease of generating QR codes and the ability to quickly modify them to link to malicious websites make them an attractive tool for attackers. This risk is amplified by users' trust in QR codes and their tendency to scan them without verifying the source or destination.

Out of 25 studies reviewed on techniques for detecting QR code attacks, 16 specifically focused on identifying malicious URLs embedded within QR codes, as summarized in Table 5. These malicious URLs redirect users to harmful websites for phishing sensitive information or downloading malware. Based on these insights, it can be concluded that embedding malicious URLs in QR codes is the most common and preferred attack strategy.

Table 5: Techniques for detecting QR code attacks

Ref.	Technique used	Protect against	Limitations	Pros
[28]	Secret sharing	Data privacy	Lack of database for storage	Ensure secure data transfer
[19]	QRCS-Digital signature	Integrity/authenticity of QR codes	QRCS is not convenient to use for both companies and everyday users by separating the application into security levels needed by the interested party	Ensures the integrity and validity of QR codes

(Continued)

Table 5 (continued)

Ref.	Technique used	Protect against	Limitations	Pros
[19]	AI-Decision tree classifier	Malicious URL	Computationally expensive and slow to train with a large dataset	Ability to recognize fraudulent URLs
[7]	PhishTank API/Browsing API	Phishing and malicious URL	Rely on continuously updated databases of known phishing site	Enhance the detection of malicious URLs
[32]	Digital signatures (ECDSA)	Web spamming	Spammers can exploit weaknesses in the implementation of digital signature algorithms or the underlying PKI infrastructure	Verify the authenticity of the QR code content and ensure its integrity
[8]	BarSec App-symmetric and asymmetric cryptographic scheme	Barcode authentication, data integrity, access control, and confidentiality	Factors like lighting, angle, and distance can affect the app's ability to read barcodes	It provides many security features, such as data integrity, confidentiality, access control, and barcode authentication, and also offers usability warnings.
[33]	Digital Signature Algorithm (DSA), sandbox, *IDS/Antivirus/firewall*	Malicious code	Compromised, expired, or poorly managed keys can undermine the entire system, allowing malicious actors to exploit vulnerabilities	Guards against privacy invasion access to personal data, and acquisition of mobile device control
[34]	Cryptographic scheme (AES-(SQR))	Decoding, phishing	Resource-intensive and may not scale well in environments with high scanning volumes	Prevent phishing, pharming, exploitation, and manipulation of information
[35]	2F-Authentication-(QRP)	Authentication	Attackers can intercept 2FA codes through MitM attacks, especially if the communication channel is insecure	Provide a secure two-factor authentication
[36]	Mutual authentication protocol	Authentication	Attackers can intercept 2FA codes through MitM attacks, especially if the communication channel is insecure	Provides better security and can distinguish between public and secret user information
[37]	Two-dimensional barcode-OTP	Authentication, phishing	Attackers can create malicious QR codes that encode phishing URLs or other harmful information, which users might scan inadvertently	Verify user identity and prevent phishing

(Continued)

Table 5 (continued)

Ref.	Technique used	Protect against	Limitations	Pros
[38]	Hybrid mechanism with the use of Quick Response code (QR code), One Time Password (OTP), and digital watermarking data hiding technique	Pharming, phishing	Advanced attackers might find ways to alter or remove digital watermarks, reducing their effectiveness	Mitigate against phishing and pharming
[39]	AI-XSStudent, APP Inventor	Cross-site scripting attacks (XSS)	Wide range of payloads and techniques to execute malicious scripts	Able to detect malicious URLs and prevent XSS attacks
[40]	Steganography and Cryptography	Security of message in the QR code	Advanced steganalysis tools can detect hidden messages	Provide an extra layer to improve the security of messages in the QR code
[15]	QsecR App	Malicious URL	Challenges in detecting new malicious URL	Able to detect malicious URL
[41]	mQRCode	Confidentiality-private information leakage	Efficiency depends on the decryption rate	Ensure confidentiality and privacy of QR code content
[42]	Association rule mining Pat-terns.Web_URL_matcher and URLUtil.isValidURL	QR Code authentication and phishing detection, malicious URL	Not effective in detecting dynamically generated malicious URLs that do not match known patterns	Able to detect malicious URLs, and phishing and provide authentication capability
[43]	Rotating the QR code	Information privacy	Not all QR code readers or scanners may support rotating QR codes	Ensure information privacy and prevent information loss
[44]	SafeQR codes/multi-faceted defense strategy	Quishing		Prevent QR code tampering and prevent phishing
[45]	QR shield-dual machine learning (Random Forest (RF) and XGBoost algorithms)	Malicious URL	Computationally expensive with a large dataset	Detech fraudulent URLs encoded in QR codes
[46]	Anti-Malware Phishing Scanner (AMPS)-Advanced Encryption Standard	Malicious URL	False positives	Detect malicious URLs and provide data encryption
[47]	A detection method using the number of errors, a detection method using the locations of errors, and a detection method using the symmetry and locality of errors	Detection of fake QR codes	Unable to correctly detect certain stained QR codes	Detect fraudulent QR codes

(Continued)

Table 5 (continued)

Ref.	Technique used	Protect against	Limitations	Pros
[48]	Transformer-based algorithm/ vision-based approach	Verification, authorization, and auditing	Require significant computational resources	Able to authenticate human scanners
[49]	Automatic detector (ResNet50, MobilenetV3 and InceptionV3 CNN models)	Malicious QR codes	Require significant computational resources	Identify and notify users of any malicious content in QR codes
[50]	Swarm-Intelligent Squirrel-Search Optimized Artificial Neural Network (SISSO-ANN)	Malicious URL	Malicious URLs evolve rapidly with new techniques and obfuscation methods	Detect malicious URL

Due to technological improvements and the resourcefulness of malevolent actors, new dangers and exploitation techniques can emerge as QR codes are incorporated more and more into routine transactions and digital interactions. Advanced QR codes with malicious payloads, including Trojan programs or backdoors, could be made using generative AI. By adding verifiable, invisible watermarks to authentic QR codes, this can be prevented. To trick users into scanning the incorrect QR code, attackers may employ augmented reality techniques to digitally overlay harmful codes onto authentic ones in real time. Adding obvious security features to QR codes, such as logos or patterns, helps prevent this by making it more difficult to conceal or swap them out.

4.3.2 RQ2: What Techniques Exist for Detecting QR Code Attacks?

To address the second research question regarding QR code attack detection techniques, this study identified various methods proposed to combat this issue, as shown in Table 4. Among the 25 studies focused on QR code attack detection techniques, 10 studies utilized cryptographic methods, 7 studies applied artificial intelligence, 2 studies employed one-time passwords, and the remaining studies relied on approaches such as user education, two-factor authentication (2FA), multi-factor authentication (MFA), or blockchain technology. This indicates that cryptographic and artificial intelligence techniques are the most commonly favored approaches for mitigating QR code attacks.

When used properly, cryptographic methods can be quite successful in reducing the hazards related to QR code assaults. Sensitive information encoded in QR codes can be encrypted. The data can only be accessed by those who possess the decryption key. Furthermore, including a digitally signed payload inside the QR code guarantees that the information is authentic and unaltered.

By identifying rogue QR codes, stopping phishing efforts, and enhancing authentication systems, machine learning (ML) techniques are being employed more and more to improve QR code security. Techniques such as CNN, Random forest, XGBoost, and Decision tree classifiers are some of the common machine learning techniques that are employed. However, the complexity of the models, the nature of the tasks they handle, and the resources needed for training and inference are some of the reasons why artificial intelligence (AI)-based approaches can be computationally costly.

The proposed techniques protect users against phishing, SQL command injection, pharming, and XSS attacks, propagation of malicious URLs, and loss of data privacy and confidentiality.

The Table 6 below shows the accuracy of different techniques for malware detection in QR codes.

Table 6: Techniques for malware detection accuracy

Ref.	Accuracy (%)
[15]	93.5
[19]	90.243
[41]	98.6
[47]	96.8
[49]	98.13
[50]	93.80

5 Conclusion and Recommendations

A QR code, a two-dimensional barcode, can be read using a smartphone or a device equipped with a QR code reader. These codes can be scanned using a smartphone camera or a dedicated QR code reader app. QR codes serve various purposes, such as accessing websites, launching mobile applications, viewing restaurant menus, making payments, placing phone calls, connecting to Wi-Fi without a password, sending emails, adding contact information, and more. Their simple design and ease of use have contributed to their widespread global adoption. However, this convenience also exposes users to risks, as they cannot easily discern the content hidden within a QR code before scanning it. Identified security threats include QR-in-QR attacks, payment fraud, counterfeiting, and information leakage. Additionally, malicious actors can exploit QR codes for phishing, pharming, malware distribution, cross-site scripting (XSS), SQL/command injection, and attacks targeting QR reader applications.

Several techniques have been proposed to mitigate QR code attacks, including cryptographic methods, machine learning, artificial intelligence, two-factor authentication, one-time passwords, and mutual authentication schemes. Users must remain cautious when scanning QR codes and take measures to verify their authenticity. While QR codes provide a convenient and versatile way to share information, process payments, and enhance business operations, it is important to recognize and address the potential security risks they pose.

For future work, this study suggests enhancement of QR code security by developing advanced neural networks capable of identifying malicious QR codes based on subtle visual or encoded patterns and employing blockchain technology to ensure the integrity and authenticity of QR codes through an immutable record of legitimate codes. Together, these technologies will address both detection and prevention, providing a comprehensive security framework.

Recommendations

To mitigate the menace of QR code attacks, this study recommends that more research is needed to develop automated detection techniques that can authenticate QR codes and detect malicious URLs or malware in real-time.

Additionally, to ensure the secure implementation of QR codes, companies and service providers can adopt a comprehensive set of regulations and guidelines that address design, deployment, user education, and ongoing monitoring. For instance encrypt sensitive data encoded in QR codes to prevent unauthorized access, implement built-in URL safety checks in scanning apps to warn users about untrusted or suspicious domains, and embed digital watermarks or invisible security features in QR codes to detect tampering. Users

should be advised to scan QR codes only from trusted sources and verify the destination before taking further action.

To reduce the risk of QR code malware and phishing attacks, the study suggests adopting safe user practices. These include using QR scanning applications equipped with security features, such as detecting malicious URLs or prompting user confirmation before accessing links. Users should avoid apps that automatically perform actions like opening links or downloading files without their input. It is also recommended to preview the link embedded in a QR code before opening it and to refrain from sharing sensitive information through QR codes.

Acknowledgement: Our heartfelt appreciation goes to all the individuals and institutions that reviewed and provided constructive feedback on this research paper. Your valuable input helped improve the quality and rigor of our work. Without the collective efforts and support of all these individuals and organizations, this research paper would not have been possible. Thank you all for your invaluable contributions.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: The authors confirm their contribution to the paper as follows: study conception and design: David Njuguna; analysis and interpretation of results: David Njuguna and John Ndia; draft manuscript preparation: David Njuguna and John Ndia. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The authors confirm that the data supporting the findings of this study are available within the article.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

Supplementary Materials: The supplementary material is available online at https://doi.org/10.32604/jcs.2025.059398.

References

1. Focardi R, Luccio FL, Wahsheh HAM. Security threats and solutions for two-dimensional barcodes: a comparative study. In: Daimi K, editor. Computer and network security essentials. Cham: Springer International Publishing; 2018. p. 207–19. doi:10.1007/978-3-319-58424-9_12.
2. ISO/IEC 18004:2015(en). Information technology—Automatic identification and data capture techniques—QR Code bar code symbology specification. [cited 2025 Jan 14]. Available from: https://www.iso.org/obp/ui/#iso:std:iso-iec:18004:ed-3:v1:en.
3. QR Code development story|Technologies|DENSO WAVE. [cited 2025 Jan 14]. Available from: https://www.denso-wave.com/en/technology/vol1.html.
4. Lerner A, Saxena A, Ouimet K, Turley B, Vance A, Kohno T, et al. Analyzing the use of quick response codes in the wild. In: Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services; 2015 May; Florence Italy: ACM. p. 359–74. doi:10.1145/2742647.2742650.
5. Thompson N, Lee K. Are QR codes the next phishing risk? ACS Inf. Age SepOct. 2012:36–7.
6. Zhou Y, Jiang X. Dissecting android malware: characterization and evolution. In: 2012 IEEE Symposium on Security and Privacy; 2012; IEEE. p. 95–109.
7. Yao H, Shin D. Towards preventing QR code based attacks on android phone using security warnings. In: Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security; 2013 May; Hangzhou China: ACM. p. 341–6. doi:10.1145/2484313.2484357.
8. Wahsheh HA, Luccio FL. Security and privacy of QR code applications: a comprehensive study, general guidelines, and solutions. Information. 2020;11(4):217. doi:10.3390/info11040217.

9. Chiew KL, Yong KSC, Tan CL. A survey of phishing attacks: their types, vectors, and technical approaches. Expert Syst Appl. 2018;106:1–20. doi:10.1016/j.eswa.2018.03.050.
10. Yong KS, Chiew KL, Tan CL. A survey of the QR code phishing: the current attacks and countermeasures. In: 2019 7th International Conference on Smart Computing & Communications (ICSCC); 2019; IEEE. p. 1–5.
11. Song J, Gao K, Shen X, Qi X, Liu R, Choo K-KR. QRFence: a flexible and scalable QR link security detection framework for Android devices. Future Gener Comput Syst. 2018;88:663–74. doi:10.1016/j.future.2018.05.082.
12. Krombholz K, Frühwirt P, Rieder T, Kapsalis I, Ullrich J, Weippl E. QR code security-how secure and usable apps can protect users against malicious QR codes. In: 2015 10th International Conference on Availability, Reliability and Security; 2015; IEEE. p. 230–7.
13. Kieseberg P, Leithner M, Mulazzani M, Munroe L, Schrittwieser S, Sinha M, et al. QR code security. In: Proceedings of the 8th International Conference on Advances in Mobile Computing and Multimedia; 2010 Nov; Paris France: ACM. p. 430–5. doi:10.1145/1971519.1971593.
14. Krombholz K, Frühwirt P, Kieseberg P, Kapsalis I, Huber M, Weippl E. QR code security: a survey of attacks and challenges for usable security. In: Tryfonas T, Askoxylakis I, editors. Human aspects of information security, privacy, and trust. Cham: Springer International Publishing, Lecture Notes in Computer Science; 2014. Vol. 8533, p. 79–90. doi:10.1007/978-3-319-07620-1_8.
15. Rafsanjani AS, Kamaruddin NB, Rusli HM, Dabbagh M. Qsecr: secure QR code scanner according to a novel malicious URL detection framework. IEEE Access. 2023;11:92523–39. doi:10.1109/ACCESS.2023.3291811.
16. Yuan T, Wang Y, Xu K, Martin RR, Hu S-M. Two-layer QR codes. IEEE Trans Image Process. 2019;28(9):4413–28. doi:10.1109/TIP.2019.2908490.
17. Averin A, Zyulyarkina N. Malicious QR-Code threats and vulnerability of blockchain. In: 2020 Global Smart Industry Conference (GloSIC); 2020; IEEE. p. 82–6.
18. Chou G-J, Wang R-Z. The nested QR code. IEEE Signal Process Lett. 2020;27:1230–4. doi:10.1109/LSP.2020.3006375.
19. Al-Zahrani MS, Wahsheh HAM, Alsaade FW. Secure real-time artificial intelligence system against malicious QR code links. Secur Commun Netw. 2021 Dec;2021:1–11. doi:10.1155/2021/5540670.
20. Mavroeidis V, Nicho M. Quick response code secure: a cryptographically secure anti-phishing tool for QR code attacks. In: Rak J, Bay J, Kotenko I, Popyack L, Skormin V, Szczypiorski K, editors. Computer network security. Cham: Springer International Publishing, Lecture Notes in Computer Science; 2017. vol. 10446, p. 313–24. doi:10.1007/978-3-319-65127-9_25.
21. Kieseberg P, Schrittwieser S, Leithner M, Mulazzani M, Weippl E, Munroe L, et al. Malicious pixels using QR codes as attack vector. In: Khalil I, Mantoro T, editors. Trustworthy ubiquitous computing. Paris: Atlantis Press, Atlantis Ambient and Pervasive Intelligence; 2012. Vol. 6, p. 21–38. doi:10.2991/978-94-91216-71-8_2.
22. Mounika G, Rani MGN, Ramakrishna VM. A blockchain and qr code based malicious transactions analysis and detecting the vulnerability. Int J Eng Res Sci Technol. 2023;19(2):27–32.
23. Kharraz A, Kirda E, Robertson W, Balzarotti D, Francillon A. Optical delusions: a study of malicious QR codes in the wild. In: 2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks; 2014; IEEE. p. 192–203.
24. Thompson N, Lee K. Information security challenge of QR codes. J Digit Forensics Secur Law. 2013;8(2):2. doi:10.15394/jdfsl.2013.1143.
25. Dabrowski A, Krombholz K, Ullrich J, Weippl ER. QR inception: barcode-in-barcode attacks. In: Proceedings of the 4th ACM Workshop on Security and Privacy in Smartphones & Mobile Devices; 2014 Nov; Scottsdale Arizona USA: ACM. p. 3–10. doi:10.1145/2666620.2666624.
26. Focardi R, Luccio FL, Wahsheh HA. Usable security for QR code. J Inf Secur Appl. 2019;48:102369. doi:10.1016/j.jisa.2019.102369.
27. Ibrahim H, Karabatak S, Abdullahi AA. A study on cybersecurity challenges in e-learning and database management system. In: 2020 8th International Symposium on Digital Forensics and Security (ISDFS); 2020; IEEE. p. 1–5.
28. Gupta S, Gupta BB. Cross-Site Scripting (XSS) attacks and defense mechanisms: classification and state-of-the-art. Int J Syst Assur Eng Manage. 2017 Jan;8(S1):512–30. doi:10.1007/s13198-015-0376-0.

29. Jin X, Hu X, Ying K, Du W, Yin H, Peri GN. Code injection attacks on HTML5-based mobile apps: characterization, detection, and mitigation. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security; 2014 Nov; Scottsdale Arizona USA: ACM. p. 66–77. doi:10.1145/2660267.2660275.
30. Alaca Y, Çelik Y. Cyber attack detection with QR code images using lightweight deep learning models. Comput Secur. 2023;126:103065. doi:10.1016/j.cose.2022.103065.
31. Chuang J-C, Hu Y-C, Ko H-J. A novel secret sharing technique using QR code. Int J Image Process. 2010;4(5):468–75. doi:10.26438/ijcse/v7i6.882887.
32. Razzak F. Spamming the Internet of Things: a possibility and its probable solution. Procedia Comput Sci. 2012;10:658–65. doi:10.1016/j.procs.2012.06.084.
33. Bani-Hani RM, Wahsheh YA, Al-Sarhan MB. Secure QR code system. In: 2014 10th International Conference on Innovations in Information Technology (IIT); 2014 Nov. p. 1–6. doi:10.1109/INNOVATIONS.2014.6985772.
34. Goel N, Sharma A, Goswami S. A way to secure a QR code: SQR. In: 2017 International Conference on Computing, Communication, and Automation (ICCCA); 2017; IEEE. p. 494–7.
35. Shamal S, Monika K, Neha N. Secure authentication for online banking using QR code. IJETAE-Int J Emerg Technol Adv Eng. 2014.
36. Huang C-T, Zhang Y-H, Lin L-C, Wang W-J, Wang S-J. Mutual authentications to parties with QR-code applications in mobile systems. Int J Inf Secur. 2017 Oct;16(5):525–40. doi:10.1007/s10207-016-0349-6.
37. Gandhi A, Salunke B, Ithape S, Gawade V, Chaudhari S. Advanced online banking authentication system using one-time passwords embedded in QR code. Int J Comput Sci Inf Technol. 2014;5(2):1327–9. doi:10.57159/gadl.jcmm.3.3.240121.
38. Thomas J, Goudar RH. Multilevel authentication using QR code based watermarking with mobile OTP and Hadamard transformation. In: 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI); 2018; IEEE. p. 2421–5.
39. Rodriguez G, Torres J, Flores P, Benavides E, Nuñez-Agurto D. XSStudent: proposal to avoid cross-site scripting (XSS) attacks in universities. In: 2019 3rd Cyber Security in Networking Conference (CSNet); 2019; IEEE; p. 142–9.
40. Jain R. Enhancing the security of message in the QR Code using a Combination of Steganography and Cryptography [Ph.D. thesis]. Dublin, National College of Ireland; 2019.
41. Pan H, Chen Y-C, Yang L, Xue G, You C-W, Ji X. mQRCode: secure QR Code using nonlinearity of spatial frequency in light. In: The 25th Annual International Conference on Mobile Computing and Networking; 2019 Oct; Los Cabos Mexico: ACM. p. 1–18. doi:10.1145/3300061.3345428.
42. Ismail S, Alkawaz MH, Kumar AE. Quick response code validation and phishing detection tool. In: 2021 IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE); 2021; IEEE. p. 261–6.
43. Garnaik SS, Kim Y, Ryoo J. SQR: secure QR transaction with randomized rotation. In: 2022 13th International Conference on Information and Communication Technology Convergence (ICTC); 2022; IEEE. p. 1697–702.
44. Bekavac LJL, Mayer S, Strecker J. QR-code integrity by design. In: Extended Abstracts of the CHI Conference on Human Factors in Computing Systems; 2024 May; Honolulu HI USA: ACM. p. 1–9. doi:10.1145/3613905.3651006.
45. Almousa H, Almarzoqi A, Alassaf A, Alrasheed G, Alsuhibany AS. QR shield: a dual machine learning approach towards securing QR codes. Int J Comput Digit Syst. 2024;16(1):887–98. doi:10.12785/ijcds/160164.
46. Hegde N, Bharti R, Sur. Anti-malware phishing. QR Scanner. 2018;3(5):346–51. doi:10.48084/etasr.7777.
47. Ohigashi T, Kawaguchi S, Kobayashi K, Kimura H, Suzuki T, Okabe D, et al. Detecting fake QR codes using information from error-correction. J Inf Process. 2021;29:548–58. doi:10.2197/ipsjjip.29.548.
48. Wang R, Huang L, Madden K, Wang C. Enhancing QR code system security by verifying the scanner's gripping hand biometric. In: Proceedings of the 17th ACM Conference on Security and Privacy in Wireless and Mobile Networks; 2024 May; Seoul Republic of Korea: ACM. p. 42–53. doi:10.1145/3643833.3656128.
49. Minocha A, Goyal A, Gandhi R. Recognition of valid QR codes with machine learning. In: 2024 IEEE 13th International Conference on Communication Systems and Network Technologies (CSNT); 2024; IEEE. p. 724–30.
50. Marappan S, Marappan H. Enhancing malicious QR code link detection with swarm-intelligent squirrel-search optimized ANN. In: 2023 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES); 2023; IEEE. p. 1–8.