Hindawi
Security and Communication Networks
Volume 2019, Article ID 8105787, 8 pages
https://doi.org/10.1155/2019/8105787

*Research Article*

# A QR Code Secret Hiding Scheme against Contrast Analysis Attack for the Internet of Things

**Qinglan Zhao,[1,2] Shuntong Yang,[1] Dong Zheng,[1,3] and Baodong Qin[1]**

[1]*National Engineering Laboratory for Wireless Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, China*
[2]*Key Laboratory of Applied Mathematics (Putian University), Fujian Province University, Fujian, Putian 351100, China*
[3]*Westone Cryptologic Research Center, Beijing 100070, China*

Correspondence should be addressed to Dong Zheng; zhengdong_xupt@sina.com

Received 14 March 2019; Accepted 9 May 2019; Published 3 July 2019

Guest Editor: Fagen Li



Due to the advantages of larger content and error correction capability, quick response (QR) code is commonly used as a tagging technology for the Internet of Things (IoT) recently. However, the cover message of QR code can be easily decoded by a QR code reader, which causes the security and privacy of the cover message to raise the important issues. In this paper we present a new secret hiding scheme based on QR code. The proposed scheme has low computational complexity and is suitable for low-power devices in IoT systems because of utilizing the error correction property of QR code to hide secret information. The proposed scheme hides the secret information without changing the cover message of QR code and the user can get the cover message by using a general scanner, which contributes to reducing attacker's curiosity. The hidden secret information can be read by a special scanner with the help of the user key. One thing which is better than other known schemes is that the proposed scheme can resist contrast analysis attack. In addition, experimental results show the proposed scheme has feasibility, low computational complexity, and high hiding payload.

## 1. Introduction

The Internet of Things (IoT) interconnects physical and digital objects that are identifiable and may interact with each other and with users. These objects, each with its own identity, are well beyond only computers and they are our cars, luggage, household applicants, humans, and so on. This was made possible by different tagging technologies like radio frequency identification (RFID) and two-dimensional (2D) barcode which allow physical objects to be identified and refer over the IoT. Due to the less complexity and a cheap solution of 2D barcodes, they have become popular for building an IoT system. Quick response (QR) code, as a 2D barcode with the advantages of larger QR content and error correction capability, is commonly used recently.

QR code can store rich information including text, URL link, and other types of data. They can be used as a data carrier to allow users to access the system more conveniently in an IoT system. However, the cover message of the QR code can be easily read by a barcode reader [1], such as a mobile phone with a camera. This may lead to leaking of privacy. Another important issue is the security of QR code when we use a QR code to communicate secret information.

The traditional method of encrypting secret information into cipher texts makes it impossible for an attacker to obtain secret information, thereby achieving confidentiality [2–4]. This method requires a lot of computation and is sometimes not suitable for the objects of IoT system. It also clearly points out what is important information and easily attracts the attention of the attacker to increase the possibility of being attacked [5]. Information hiding technologies based on QR code have emerged to solve these new problems. The image hiding schemes [6–8] are mainly to convert the secret into a QR code tag and then embed the secret QR code into the image. However, these schemes require complicated image processing operations to recover the hidden QR code. Most watermarking algorithms, which use Discrete Cosine Transform, Discrete Wavelet Transform, and Discrete Fourier Transform algorithms to hide the watermark in the QR code [9–12], have high computational complexity and limited

hidden information caused by the length and width of QR code. Therefore, due to the high computational complexity of these methods, they are not suitable for low-power mobile devices.

To meet the demands of applications of QR code to low-power mobile devices in IoT systems, some schemes have been presented which used the code characteristic of the QR code to hide secret [13–17]. Chiang et al. proposed a scheme [14] to hide the secret information being confused by the pseudo-random binary stream generated by the user key. To increase the hiding payload, [16] proposed a data hiding method which is an extended version of [13]. However, these methods can not resist the contrast analysis attack. Under such an attack scenario, the attacker can contrast the codewords of QR codes which have different cover message and hide the same secret information with the same user key. These methods insert some data related to the secret information into the original codewords and make positions of secret message unchanged when the secret and key do not change. The attacker can get the data which are the same part of these codewords. Even the attacker can not recover the secret from the data they got without the key, they can create a new QR code with embedding these data which hide secret information. When these secret schemes are used for copyright protection, by this method the attacker can forge copyright information containing the legal copyright information.

In order to resist the contrast analysis attack, we design a new QR code secret hiding scheme. The proposed scheme makes the changed codewords of original QR code related to the cover message using the simple XOR operation. Compared with original QR code, QR codes have different changed codewords if they have the different cover message and hide the same secret information with the same key. So the attacker can not find the same data related to the secret information and key through the contrast analysis attack. In addition, the proposed scheme utilizes the biggest error correction ability of QR code to resist brute force attack. With higher security than the known schemes, the proposed scheme has the low computational complexity and high hiding payload.

The paper is organized as follows. Section 2 introduces QR code technique. The proposed secret hiding scheme is described in Section 3. The simulation, performance comparisons, and security analysis are discussed in Section 4. Finally, Section 5 concludes the paper.

## 2. The Technology of QR Code

QR code is one of the most popular 2D barcodes [18]. It consists of white and black square modules which are equal to the binary values 0 and 1. Figure 1 depicts an instance of QR code symbol. The number of modules increases with QR code version. There are 40 QR code standard versions among which Version 1 has the smallest 21 × 21 modules and Version 40 has the largest 177 × 177 modules. The data payload becomes larger as the version evolves. There is 208 data modules in Version 1 and 29648 data modules in Version 40. The message bit stream shall be divided into codewords. All codewords are 8 bits in length.

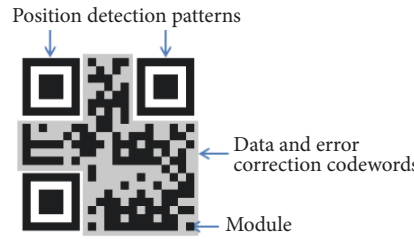



Figure 1: Basic structure of a QR barcode.

Table 1: Error correction levels.

| Error Correction Level | Recovery Capacity %(approx.) |
|---|---|
| L | 7 |
| M | 15 |
| Q | 25 |
| H | 30 |

To achieve the recovery ability, the error correction algorithm has been used in QR code to generate a series of error correction codewords which are added to the data codeword sequence. The error up to 30% can be corrected. Each version has four error correction levels L, M, Q, and H as shown in Table 1. Depending on the version and error correction level, the data codeword sequence is subdivided into one or more blocks, to each of which the error correction algorithm shall be applied separately to the data codeword. To show this, the error correction characteristics of QR code of Version 1, 20, and 40 are listed in Table 2, where $c$ is the total number of codewords, $k$ is the number of data codewords, and $r$ is the number of error correction capacity. For Version 20 with error correction level L, as an example, 1085 codewords are divided into 8 blocks in which 3 blocks apply error correction codewords (135,107,14) and 5 blocks apply error correction codewords (136,108,14).

The process to construct a QR code is structured into seven steps.

*Step 1* (data analysis). The input data stream is analyzed to identify the variety of different characters to be encoded and the version and error correction level are selected.

*Step 2* (data encodation). Data characters are converted to a bit stream which is split into 8-bit codewords.

*Step 3* (error correction coding). The codeword sequence is divided into the required number of blocks and the error correction codewords are calculated for each block with being appended to the end of the data codeword sequence.

*Step 4* (structure final messages). The final sequence is assembled by taking data and error correction codewords from each block in turn.

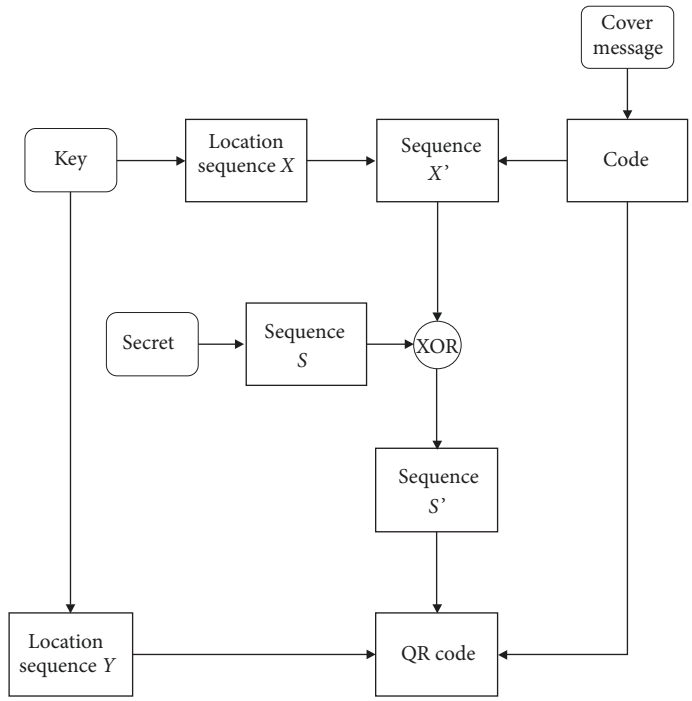



Figure 2: Secret hiding procedure.

*Step 5* (module placements in matrix). The codeword modules are placed in the matrix together with other patterns.

*Step 6* (masking). To optimize the dark/light module balance and minimize the occurrence of undesirable patterns, the masking pattern is applied.

*Step 7* (format and version information). The format and version information are created.

## 3. The Proposed Scheme

The proposed scheme utilizes the code characteristic of QR code to hide secret information. It includes two procedures, secret hiding procedure and secret extraction procedure, whose flowcharts are shown in Figures 2 and 3. Secret hiding procedure is integrated with the QR code generation for a cover message and secret extraction procedure is integrated with the QR code scanning, as we will describe in detail.

*3.1. Secret Hiding Procedure.* For the secret information to be hidden, its code $MC$ can be given according to the coding principle of the QR code. $MC$ is expanded to a longest sequence $S$ whose length is related to error correction of the QR code version. Suppose $r$ be the number of error correction capacity. The length of $S$ is $8r$. In order to resist QR contrast attack, $S$ will be confused by a sequence related to the cover message. The following are details of secret hiding procedure.

*Step 1.* Perform the normal QR code encoding procedure for the secrete information and cover message until their codewords are generated.

*Step 2.* Use user's key $ks$ as input to generate a location sequence $Y = [Y_0, Y_1, \ldots, Y_{8r-1}]$, where $Y_i = (m, n)$, such that $m \in C'$ with $C'$ being a subset of $\{0, 1, \ldots, c-1\}$ and $|C'| = r$, $0 \le n \le 7$ for all $0 \le i \le 8r-1$.

*Step 3.* Use user's key $ks$ as input to generate a sequence $X = [X_0, X_1, \ldots, X_{8r-1}]$ with $X_i = (t, s)$, $0 \le t \le c-1, 0 \le s \le 7$ such that the $t$th codeword is one of data codewords for all $0 \le i \le 8r-1$.

*Step 4.* Let $l$ be the length of the secret codewords $MC$ and the length of the code of $l$ be $q$. Sequence $0 \ldots 0$ with length $l = r * 8 - q$ is added to code $MC$, and then the code of $l$ is added at the end. The resulting sequence is $S = [S_0, S_1, \ldots, S_{8r-1}]$.

*Step 5.* According to $X$, find the data $X'_i$ of the block on the position $X_i = (t, s)$, that is, the $s$th data of the $t$th codeword, for all $0 \le i < 8r-1$. Then get a sequence $X' = [X'_0, X'_1, \ldots, X'_{8r-1}]$.

*Step 6.* Generate a sequence $S' = [S'_0, S'_1, \ldots, S'_{8r-1}]$ with $S'_i = S_i \oplus X'_i$ for $0 \le i \le 8r-1$.

*Step 7.* Embed the sequence $S'$ into the cover message codewords according to the position sequence $Y$. For $Y_i = (m, n)$ with $0 \le i \le 8r-1$, look for the $n$th bit of the $m$th block in the cover message codewords and replace it with $S'_i$.

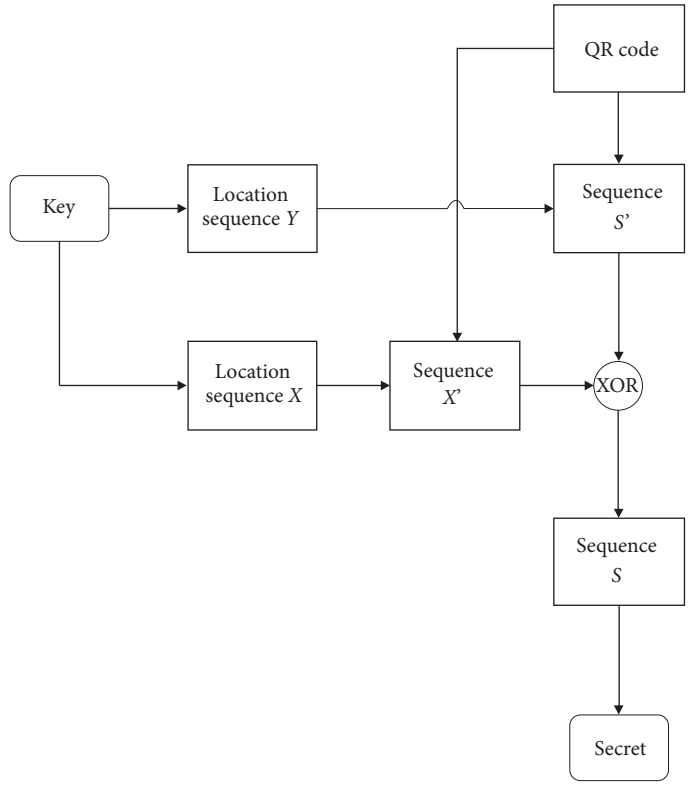



Figure 3: Secret extraction procedure.

*Step 8.* Continue the process of encoding the QR code with hidden secret information.

*3.2. Secret Extraction Procedure.* To extract the secret information, we design the special scanner in which the secret extraction is integrated with scanning process. The common scanner only can read the cover message by scanning the QR code. Using the special scanner, the authorized user can get the cover message and the secret information with inputting the key *ks*. Secret extraction procedure includes the following steps.

*Step 1.* Scan QR code to get data block information before error correction.

*Step 2.* Use user's key *ks* as input to generate a location sequence $Y$.

*Step 3.* According to the location sequence $Y$, for any $Y_i = (m, n)$ with $0 \le i \le 8r - 1$, find the $n$th bit data $S'_i$ of the $m$th codeword in the QR code and get the sequence $S' = [S'_0, S'_1, \ldots, S'_{8r-1}]$.

*Step 4.* Finish error correction and use user's key as input to generate a sequence $X$.

*Step 5.* According to $X$, find the data $X'_i$ on the position $X_i = (t, s)$ of the block and get the sequence $X' = [X'_0, X'_1, \ldots, X'_{8r-1}]$ corresponding with $X = [X_0, X_1 \ldots, X_{8r-1}]$, $0 \le i \le 8r - 1$.

*Step 6.* Generate sequence $S = [S_0, S_1, \ldots, S_{8r-1}]$ with $S_i = S'_i \oplus X'_i$ for $0 \le i \le 8r - 1$.

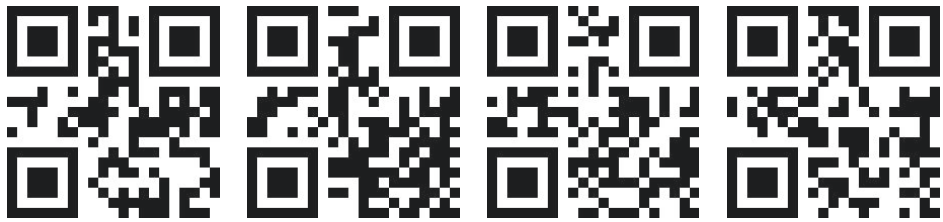
*Step 7.* Calculate the length $l$ of the secret according to the last $q$th bits and then get the secret codewords $[S_0, S_1, \ldots, S_{l-1}]$.

## 4. Simulation Results and Analysis

To assess the feasibility and suitability of the scheme, we implement the proposed secret hiding scheme using Python language which has a powerful operation library of QR code.

*4.1. Experimental Results and Practicability.* The results of the proposed scheme for the 1-H QR versions are shown in Figure 4. Figure 4(a) is the original QR code image with the cover message "Data". Figure 4(b) is the QR code image which has the same cover message with Figure 4(a) and hides the secret message "Secret" with the user key "google". Figures 4(c) and 4(d) hide the same secret and use the same user key as Figure 4(b), but have the different cover message "Escher" and "Linux".

To show the hiding procedure, we introduce *QRC* which denote the coding function of QR code including step 1 and step 2 of constructing QR code and *QRCS* denote the coding



(a) (b) (c) (d)

Figure 4: The results of 1-H QR code in different cases.

Table 2: Error correction characteristics of Versions 1, 20, and 40.

| Version | Total number of codewords | Error correction level | Number of error correction codewords | Number of error correction blocks | Error correction code per block ($c$,$k$,$r$) |
|---|---|---|---|---|---|
| 1 | 26 | L | 7 | 1 | (26,19,2) |
| | | M | 10 | 1 | (26,16,4) |
| | | Q | 13 | 1 | (26,13,6) |
| | | H | 17 | 1 | (26,9,8) |
| 20 | 1085 | L | 224 | 3 | (135,107,14) |
| | | | | 5 | (136,108,14) |
| | | M | 416 | 3 | (67,41,13) |
| | | | | 13 | (68,42,13) |
| | | Q | 600 | 15 | (54,24,15) |
| | | | | 5 | (55,25,15) |
| | | H | 700 | 15 | (43,15,14) |
| | | | | 10 | (44,16,14) |
| 40 | 3706 | L | 750 | 19 | (148,118,15) |
| | | | | 6 | (149,119,15) |
| | | M | 1372 | 18 | (75,47,14) |
| | | | | 31 | (76,48,14) |
| | | Q | 2040 | 34 | (54,24,15) |
| | | | | 34 | (55,25,15) |
| | | H | 2430 | 20 | (45,15,15) |
| | | | | 61 | (46,16,15) |

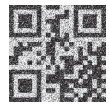
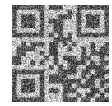
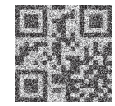
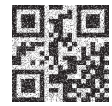
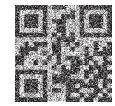
function of QR code with secret hidden. For simplicity, we use a decimal number to represents an 8-bit binary sequence. In Figure 4(a), "Data" is the cover message of the QR code. With using the mask type 7, we get Q$RC$( "Data") as follows: $QRC$(Data) = [64, 68, 70, 23, 70, 16, 236, 17, 236, 97, 53, 255, 172, 71, 43, 105, 94, 82, 129, 51, 201, 118, 131, 139, 97, 120].

In Figure 4(b), the cover message of the QR code is "Data", the user key is "google", the secret message is "Secret", and the mask type is 0. According to secret hiding procedure, we get $S = [115, 101, 99, 114, 101, 116, 0, 6]$. $S' = [155, 75, 232, 240, 129, 46, 118, 94]$. $Y = [(18, 6), (18, 0), (6, 5), (18, 2), (23, 3), (15, 4), (18, 1), (6, 6), (4, 3), (15, 6), (7, 1), (7, 0), (25, 2), (23, 5), (6, 1), (25, 3), (23, 1), (18, 4), (4, 4), (12, 3), (18, 7), (4, 5), (7, 2), (25, 1), (6, 3), (23, 0), (15, 5), (12, 6), (4, 2), (15, 3), (23, 2), (25, 7), (25, 6), (23, 6), (6, 7), (7, 4), (12, 4), (4, 1), (12, 7), (7, 6), (18, 3), (18, 5), (4, 7), (25, 0), (4, 6), (7, 7), (7, 5), (25, 4), (12, 0), (15, 1), (15, 2), (6, 2), (7, 3), (23, 4), (15, 0), (23, 7), (12, 2), (12, 1), (6, 0), (12, 5), (4, 0), (6, 4), (25, 5), (15, 7)]$. $QRCS$(Data)= [64, 68, 70, 23, 139, 16, 122, 7, 236, 97, 53, 255, 70, 71, 43, 230, 94, 82, 107, 51, 201, 118, 131, 216, 97, 54].

Comparing $QRC$("Data") and $QRCS$("Data"), we get that the positions of changed codewords are $[4, 6, 7, 12, 15, 18, 23, 25]$, in which the numbers such as 4 are position numbers of the codewords with underline.

Now we turn to Figure 4(c). In Figure 4(c), the cover message of the QR code is "Escher". The user key and the secret to be hidden are the same as in Figure 4(b) and the mask type is 0. $QRC$(Escher) = [64, 100, 87, 54, 54, 134, 87, 32, 236, 147, 246, 165, 169, 32, 81, 187, 30, 0, 51, 111, 72, 5, 181, 71, 33, 17]. $S$ and $Y$ are the same as in Figure 4(b). But $S' = [121, 113, 225, 22, 236, 220, 0, 10]$ is different from Figure 4(b). $QRCS$("Escher") = [64, 100, 87, 54, 202, 134, 7,

Table 3: The revealed results of various situations.

| Variance | 0.1 | 0.3 | 0.5 | 0.7 |
|---|---|---|---|---|
| *Gaussian noise* |  |  |  |  |
| QR content | Readable | Readable | Readable | Readable |
| Secret | Decodable | Decodable | Decodable | Decodable |
| Noise ratio | 10% | 30% | 50% | 70% |
| *Salt&Pepper noise* |  |  |  |  |
| QR content | Readable | Readable | Readable | Readable |
| Secret | Decodable | Decodable | Decodable | Decodable |
| Compression ratio | 15% | 30% | 40% | 50% |
| *Jpg compression* |  |  |  |  |
| QR content | Readable | Readable | Readable | Readable |
| Secret | Decodable | Decodable | Decodable | Decodable |
| broken's blocks | 1 | 2 | 4 | 8 |
| *Image damage* |  |  |  |  |
| QR content | Readable | Readable | Readable | Readable |
| Rotation angle | 45° | 90° | 180° | 270° |
| *Rotation* |  |  |  |  |
| QR content | Readable | Readable | Readable | Readable |
| Secret | Decodable | Decodable | Decodable | Decodable |

193, 236, 147, 246, 165, 10, 32, 81, 18, 30, 0, 188, 111, 72, 5, 181, 114, 33, 214]. Comparing *QRC*("Escher") and *QRCS*("Escher"), the position number of the changed codewords also are [4, 6, 7, 12, 15, 18, 23, 25].

In Figure 4(d), under inputting the same key with Figure 4(b), the same secret is hidden in the QR code with the cover message "Linux" and the mask type 0. *QRC*("Linux") = [64, 84, 198, 150, 231, 87, 128, 236, 17, 7, 55, 171, 236, 178, 92, 79, 9, 79, 217, 71, 42, 143, 106, 227, 208, 123]. As the cover message is changed, $S'$ is changed to be [132, 176, 181, 199, 24, 59, 171, 189]. So the final *QRCS*("Linux") = [64, 84, 198, 150, 159, 87, 152, 220, 17, 7, 55, 171, 188, 178, 92, 185, 9, 79, 2, 71, 42, 143, 106, 225, 208, 201] which has different data on the same changed codewords with Figure 4(c).

*4.2. Schemes Performance.* Considering the noise, compression, damage, etc., caused by the propagation and printing methods in practical applications, we analyzed the performances in Gaussian noise, salt and pepper noise, JPG lossy compression, damage, and rotation of QR codes generated by the proposed scheme hidden secret. As shown in Table 3, the QR code generated by the proposed scheme can be read correctly when being subjected to various distortions.

Table 4 shows a general comparison between the related schemes [6–12, 14, 16, 19] and the proposed scheme. Unlike the conventional hiding and watermarking schemes [6–12, 19], the proposed scheme utilizes the code character of the QR code and embeds the secret into the modules of the QR code directly. Hence, the QR code with the hidden secret information can be easily scanned by barcode readers, which make it suitable to the low-power mobile device applications.

Table 5 shows the tolerant secret capacity of the proposed scheme with different QR versions and error correction levels. The proposed scheme can embed at most $2r$ secret bits into the QR tag. Here, the maximum secret capacity $r$ is decided by the version and error correction level of the QR code. For example, in QR Version 1-H, the proposed scheme can embed at most 64 bits secret into the QR tag.

*4.3. Security Analysis.* In what follows we discuss the resistance of the proposed scheme to two QR code attacks.

Table 4: Comparison of related QR code schemes.

| Methods | [6, 19] | [7, 8] | [9, 11, 12] | [10] | [14, 16] | Proposed |
|---|---|---|---|---|---|---|
| Applications | Image hiding | Image hiding | Watermarking | Watermarking | Secret hiding | Secret hiding |
| Embedding domain | Frequency | Spatial | Frequency | Spatial | Spatial | Spatial |
| Computational complexity | High | Low | High | Low | Low | Low |
| Module-based | No | No | No | No | Yes | Yes |
| Robustness of secret | High | Low | High | High | High | High |
| Against contrast analysis attack | - | - | - | - | No | Tes |

Table 5: The QR data payload for different QR versions and error correction levels.

| Version and Error Correction Level | L | M | Q | H |
|---|---|---|---|---|
| 1 | 16 | 32 | 48 | 64 |
| 10 | 288 | 520 | 768 | 896 |
| 20 | 896 | 1664 | 2400 | 2800 |
| 30 | 1800 | 3248 | 4800 | 5760 |
| 40 | 3000 | 5488 | 8160 | 9720 |

Secret hiding payload in schemes [14–16] is not fixed; that is to say, the length of the secret to be embedded (which may be the original secret or the secret with confusion) is decided by the original secret, which result in their schemes being vulnerable to one kind of QR code attacks. Under this attack, when attackers have the information of the version of a QR code, they can generate a QR code in the same version with the same cover message and compare this QR code with the QR code embedding the secret and then get their different part including the secret information. If the payload of the hidden secret is not high, for example, in [15], secret could be leaked when being subjected to brute force attacks. The proposed scheme extends the length of the original secret to the maximum length by adding all 0 sequences to the code of the original secret. After confusion the extended secrets are embedded into the QR code. No matter how long the length of the secret is, the attacks get their different part with the longest length when they compare the original cover QR code with the new QR code embedding the secret. Hence, the scheme has the best resistance to brute force attack even if the original secret is short.

The position and confusion of secret to be embedded are decided only by the user key in schemes [14–16]. Hence, QR codes hiding the same secret with the different cover message and the same key have some same data. These same data can be exploited by contrast analysis attack. The proposed scheme achieves confusing secret using the codewords of the cover message. So when the cover message changes, the data of the changed codewords will be different. For example, as shown in Section 4.1, compared with original QR code in Figure 4(b), QR codes in Figures 4(c) and 4(d) have changed codewords of cover messages on the same position [4, 6, 7, 12, 15, 18, 23, 25] but have different data. Contrasting *QRCS*("Escher") and *QRCS*("Linux"), there are not the same codewords which will leak the secret information. Hence the attacker can not find position where the secret information was embedded by contrasting the code of the QR codes hiding the same secret information with the different cover message and the same key.

## 5. Conclusion

The QR code secret hiding scheme designed in this paper can hide up to 9720 bits of secret information as needed and does not affect the readability of the cover message. The secret information can be extracted by the authorized user with the right key in the proposed scheme. Hence when the QR code is copied by the attacker, the attacker can not extract the secret without the key. The proposed scheme has low computational complexity and high secret payload and is suitable for low-power devices. In addition, the basic point is that, unlike the other known schemes, the proposed scheme can resist contrast analysis attack, which can prevent forgery if the scheme is applied for e-ticket, copyright protection, and brand anticounterfeit in IoT systems.

## Data Availability

No data were used to support this study.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grants nos. 61472472, 61672414, and 61772418, the Natural Science Basic Research Plan in Shaanxi Province of China under Grant no. 2016JM6033, and the Key Laboratory of Applied Mathematics of Fujian Province University (Putian University) under Grant no. SX201807. Qinglan Zhao is supported by the Innovation

Ability Support Program in Shaanxi Province of China under Grant no. 2017KJXX-47.

## References


[1] D. Wave, "QR code standardization," 2003, http://www.qrcode.com/en/index.html.

[2] J. Katz, A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, New York, NY, USA, 1996.

[3] Z. Liu, X. Huang, Z. Hu, M. K. Khan, H. Seo, and L. Zhou, "On emerging family of elliptic curves to secure internet of things: ECC comes of age," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 3, pp. 237–248, 2017.

[4] M. Zhang, Y. Zhang, Y. Jiang, and J. Shen, "Obfuscating EVES algorithm and its application in fair electronic transactions in public clouds," *IEEE Systems Journal*, pp. 1–9, 2019.

[5] C.-T. Hsu and J.-L. Wu, "Hidden digital watermarks in images," *IEEE Transactions on Image Processing*, vol. 8, no. 1, pp. 58–68, 1999.

[6] W. Y. Chen and J. W. Wang, "Nested image steganography scheme using QR-barcode technique," *Optical Engineering*, vol. 48, no. 5, article no 057004, 2009.

[7] H.-C. Huang, F.-C. Chang, and W.-C. Fang, "Reversible data hiding with histogram-based difference expansion for QR code applications," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 2, pp. 779–787, 2011.

[8] S. Dey, K. Mondal, J. Nath, and A. Nath, "Advanced steganography algorithm using randomized intermediate qr host embedded with any encrypted secret message: ASA_QR algorithm," *International Journal of Modern Education and Computer Science*, vol. 4, no. 6, pp. 59–67, 2012.

[9] M. Sun, J. Si, and S. Zhang, "Research on embedding and extracting methods for digital watermarks applied to QR code images," *New Zealand Journal of Agricultural Research*, vol. 50, no. 5, pp. 861–867, 2007.

[10] M. Gao and B. Sun, "Blind watermark algorithm based on QR barcode," in *Foundations of Intelligent Systems*, vol. 122 of *Advances in Intelligent and Soft Computing*, pp. 457–462, Springer, Berlin, Germany, 2012.

[11] S. Rungraungsilp, M. Ketcham, V. Kosolvijak, and S. Vongpradhip, "Data hiding method for QR code based on watermark by compare DCT with DFT domain," in *Proceedings of the 3rd international conference on computer and communication technologies*, pp. 144–148, India, 2012.

[12] L. Li, R. Wang, and C. Chang, "A digital watermark algorithm for QR code," *International Journal of Intelligent Information Processing*, vol. 2, no. 2, pp. 29–36, 2011.

[13] P.-Y. Lin and Y.-H. Chen, "QR code steganography with secret payload enhancement," in *Proceedings of the 2016 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pp. 1–5, 2016.

[14] Y.-J. Chiang, P.-Y. Lin, R.-Z. Wang, and Y.-H. Chen, "Blind QR code steganographic approach based upon error correction capability," *KSII Transactions on Internet and Information Systems*, vol. 7, no. 10, pp. 2527–2543, 2013.

[15] P.-Y. Lin, Y.-H. Chen, E. J.-L. Lu, and P.-J. Chen, "Secret hiding mechanism using QR barcode," in *Proceedings of the International Conference on Signal-Image Technology Internet-Based Systems*, pp. 22–25, 2013.

[16] P.-Y. Lin and Y.-H. Chen, "High payload secret hiding technology for QR codes," *Eurasip Journal on Image and Video Processing*, vol. 2017, no. 1, article no 14, 2017.

[17] T. V. Bui, N. K. Vu, T. T. Nguyen, I. Echizen, and T. D. Nguyen, "Robust message hiding for QR code," in *Proceedings of the Tenth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*, pp. 520–523, IEEE, Kitakyushu, Japan, August 2014.

[18] D. Wave, "Information technology automatic identification and data capture techniques QR code bar code symbology specification," in *Proceedings of the International Organization for Standardization, ISO/IEC*, vol. 18004, 2015.

[19] C. Chung, W. Chen, and C. Tu, "Image hidden technique using QR-barcode," in *Proceedings of the Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*, pp. 522–525, Kyoto, Japan, September 2009.