Suraj Kumar Sahu / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 3, Issue 4, Jul-Aug 2013, pp.1738-1741

Encryption in QR Code Using Stegnography

Suraj Kumar Sahu*
Guided By- Mr. Sandeep Kumar Gonnade**
(Department of Computer Science, MATS University, Raipur
(Department computer science, MATS University, Raipur)

ABSTRACT

The paper present how QR Codes (commonly known as 'Quick Respond Codes') used in the field of Cryptography. QR Codes are mainly used to convey or store messages because they have higher or large storage capacity than any other normal barcodes. The paper describes the relation QR Code and cryptography. Since QR Codes have fast response time and have large storage capacity, QR Codes can be used perfectly to send encrypted data (messages) to the receiver. This method will be suitable in any business house, government sectors, and communication network to send their encrypted messages faster to the destination. Or a person can even use this method to keep his important documents, like passport number, pan-card id, and social security number, perfectly secured with him all the time, without the information getting leaked to outside world

Keywords: QR Code, Cryptography, Data Hiding, encryption, decryption, code generation;

I. INTRODUCTION

QR codes (short for Quick Response codes) is a two dimensional barcode were invented in 1994 by the Toyota Motors subsidiary Denso Wave to track vehicles and parts during the manufacturing process. The QR code consists of black modules (square dots) arranged in a square grid on a white background. The information encoded may be made up of data (numeric, alphanumeric, byte / binary, Kanji) or, through supported extensions, virtually any type of data.

A QR code is read by an imaging device, such as a camera, in a mobile phone and there a number of different barcode scanner applications such as Red Laser, Barcode Scanner and QR Scanner that can read and decode data from a QR code.

The majority of these are completely FREE, and all you have to do once you install one is to use your phone's camera to scan the barcode, which will then automatically load the encoded data for you.

QR code will be used at Advertising, Competitions, Business cards, Social networking (Face book , Twitter, etc.), Branding, Ticketing/registration, Campaign tracking, File access, Statistics, Logistics/parts tracking, Marketing, Payment, Reminders/updates the possibilities are endless.

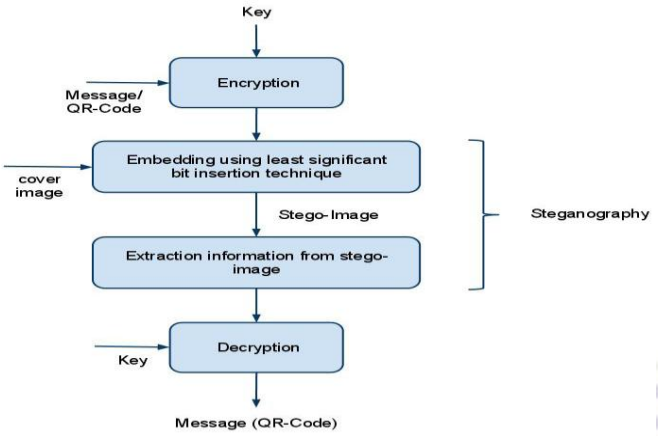
II. ENCRYPTION

In cryptography, **encryption** is the process of encoding messages (or information) in such a way that eavesdroppers or hackers cannot read it, but that authorized parties can. In an **encryption scheme**, the message or information (referred to as plaintext) is encrypted using an encryption algorithm, turning it into an unreadable cipher text (ibid.). This is usually done with the use of an encryption key, which specifies how the message is to be encoded. Any adversary that can see the cipher text should not be able to determine anything about the original message. An authorized party, however, is able to decode the cipher text using a **decryption** algorithm, that usually requires a secret decryption key, that adversaries do not have access to. For technical reasons, an encryption scheme usually needs a key-generation algorithm to randomly produce keys.

III. NEED OF ENCRYPTED QR CODE

One of the first applications that come to mind is to use encrypted QR codes on passports, driver license and other identification or even loyalty cards. Assume that every citizen will have a hidden Id residing in a secured governmental database. This hidden Id points in the database to an overt Id printed on the passport together with a name and other details. The QR code encrypts a URL and the hidden Id. An inspector scanning the QR code will get from the secured database all data linked with the hidden Id including name, birth date address and more. All the inspector has to do is compare the received data to that on the passport. With today technology even the image that should be on the passport can be sent for comparison

IV. INCRYPTION USING STEGNOGRAPHY



cover_medium + hidden_data + stego_key = stego_medium

In this context, the *cover medium* is the file in which we will hide the *hidden data*, which may also be encrypted using the *stego_key*. The resultant file is the *stego_medium* (which will, of course. be the same type of file as the cover_medium). The cover_medium (and, thus, the stego_medium) are typically image or audio files. In this article, I will focus on image files and will, therefore, refer to the *cover image* and *stego_image*.

4.1 Embedding phase

- In embedding phase embed the data or any file in the cover image that is the QR code. The first step in the embedding phase is to generate the desired QR Code. For generating the QR code choose any type of the QR image and type the data now encode the data as a QR image. Now generate our QR code that can be decode by the mobile phone using the android application

4.1.1 Encode the QR code

In embedding phase first encode the QR code of any data or information that can choose as a cover image . for encode the QR code choose the image type bmp, png, gif etc; then inter the text and now encode the QR code using the path that provide to make QR code that request to server for QR image in which loaded the image. Now given the path for encode the data. Now for save the different type of QR image gif, bmp, png etc; use the dll file that is automatic executable file . the dll file call the QR code for encode and type of image. Then save the encoded QR code.

- The Next step is choosing a cover image and chooses the file or data that embed in the QR image that can use as a cover. Now use the cover image which generated and embed the another data using stegno encryption.

  In the stegno encryption use the cover image and data that can be embed. When choose the cover image and embed the data the click the encrypt option then it save in chosen path now encrypt that is very secure.

4.1.2 Encryption using stegnography

The Next step is choosing a cover image and chooses the file or data that embed in the QR image that can use as a cover. Now use the cover image which generated and embed the another data using stegno In the stegno encryption use the cdata that can be embed. When choose the cover image and embed the data the click the encrypt option then it save in chosen path now encrypt that is very secure.

4.2 Extraction phase:decryption

In the extraction phase remove the cover image. For removing the cover image use the stegno decryption. When encrypt the QR code then send it to another party that decrypt by only our software.when decrypt the QR image then find the original file and remove the QR image that decode using smart phone but the stegno decryption can be only for using this software.

When use the decryption firstly brows the image which is already encrypt by the stegnography then save the original file and now decrypt then find the real data that is embedded and remove the cover image that decode by the phone. When decode the encrypted QR code then decode only the cover image but embedded data find after the decryption .

4.3 DLL –dynamic link library

In a nut shell, a dynamic link library (DLL) is a collection of small programs, which can be called upon when needed by the executable program (EXE) that is running. The DLL lets the executable communicate with a specific device such as a printer or may contain source code to do particular functions. An example would be if the program (exe) needs to get the free space of your hard drive. It can call the DLL file that contains the function with parameters and a call function. The DLL will then tell the executable the free space. This allows the executable to be smaller in size and not have to write the function that has already exists.

This allows any program the information about the free space, without having to write all the source code and it saves space on your hard drive as well. When a DLL is used in this fashion are also known as shared files.

The advantage of DLL files is that, because they do not get loaded into random access memory (RAM) together with the main program, space is saved in RAM. When and if a DLL file is called, then it is loaded. For example, you are editing a Microsoft Word document, the printer DLL file does not need to be loaded into RAM. If you decide to print the document, then the printer DLL file is loaded and a call is made to print.

All in all a DLL is an executable file that cannot run on its own, it can only run from inside an

executable file. This would be like having a car without an engine, where as an executable has an engine.

To do load a DLL file, an executable needs to declare the DLL function. A DLL may have many different functions in it. Then when needed the call is made with the required parameters.

4.4 Implementation scope

- **Secutity**- In Integration of QR code using stegnography generate encrypted QR code and then embed it to the QR cover image for more security. When decrypt the cover image then find the encrypted QR code that again decrypt using key now find the original QR image and then scan by using mobile.
- **Generation of QR code**-Generate the QR for all data like text, numeric, URLs, audio clip, image etc. encode the QR code for any data now find the the QR code which is use as a cover image now select the audio clip, any image, or any other data for embedding now encrypt using stegno encryption and find the QR code.
- **Storing capacity**-Integration of QR code integrate the two QR code. One QR code encode the data that can be use as a cover image now encode the another QR code for another data or imbed any file now integrate using stegnography so increase the storing capacity of data and store the maximum information.

4.5 How to embed the data in QR image

The image which have components(Red, Green and Blue) each having 8-bits, thus its possible to store 3 bits of information in every pixel, on contrary if you use a Grey scale image you can only store 1bit of information per pixel.

To understand image-based steganography, we need to understand the concept of a digital image. Images, a combination of width and height (W*H), are comprised of thousands of pixels, either 8-bit or 24-bit color combinations. With 8-bit color, there would 256 colors forming an image, due to the basic binary calculation (2^8= 256). A 24-bit color pattern is more complex and provides more color, in this case with each pixel representing 3 bytes. (Remember that 1 byte contains 8-bits and each byte represents a combination of color designated "RGB": Red, Green and Blue.) Let us suppose that an image has a size of 1200 * 800 pixels. That would be 960,000 pixels, so for 24-bit scheme where each pixel contains 3 bytes, we're talking 2,880,000 bytes; as each byte consists of 8 bits, this brings us to 23,040,000 bits. Now we have calculated that an image of 1200 * 800 pixels is based on 23040000 bits, remember this number is in decimal form. We need to convert it to binary for an in-depth analysis. So the binary of these bits would be **000101011111100100000000000**. So how do we hide a message within an image? By using the above calculation, we can easily get the binary of an image. The right side of the binary is called least significant bit (LSB) because it contains the least amount of information, while the left side top most is called most significant bit (MSB), as it contains the bulk of the information. The point is: if we replace the LSB (least significant bit) with some other bits containing other information, this will not affect the shape of the image, since what we are replacing, the LSB, did not contain much information to begin with. Let's consider an example: Suppose we have a 16 byte of data:

00110101 00101100 11001001 10010111
00001110 11001011 10011111 00010001
10010111 00000000 11001001 01010110
10101010 01001010 10010100 10000101

Now we want to hide Hi in these bytes. It's simple: first, we need to get the binary equivalent of the word Hi. We can do this by using ASCII to binary conversion. The binary of Hi is

0100100001101001

Put these bits on the LSB of the above bytes:

00110100 00101101 11001000 10010110
00001111 11001010 10011110 00010000
10010110 00000001 11001001 01010110
10101011 01001010 10010100 10000101

We have successfully hidden the word Hi into these bytes

V. HOW TO ENCRYPT QR CODES

All ways described here will use symmetrical keys, meaning that the same key used for encryption is the key used for decryption. The length of the encryption key may be at the length of both the original and error correction data. This key can be composed from a sentence or a series of non meaningful characters, and encryption is done by performing a bitwise XOR operation on both data chunks using this sequence. Redo the same operation on the encrypted message and you will get back the original message. Now after encryption, choose a number of random locations in data (no more than half of the permitted errors by the error correction level) and change the bytes in these locations randomly. After decrypting the message with a knowledgeable reader (that knows the secret key), the Reed-Solomon algorithm will correct the wrongly decrypted codeword's and the correct message will be formed.

For a version 2 QR code that contains 44 codeword's a key of length 44*8=352 bits is equivalent to a number with 106 digits. For comparison, SSL keys with 128 digits are considered to be unbreakable today. A version 3 QR code with 70 codeword's may use a key of 70 bytes equivalent to a number with 168 digits.

A harder encryption is achieved by making some errors in random places like before, this time before encryption. After that shuffling the bits of both original and error correction data in a certain order and applying to this a symmetrical key, just like before. To decrypt this you will first need to apply the symmetrical key, after that reshuffle the bits to their original position, then apply the Reed-Solomon algorithm to correct the planted errors in the message.

VI. SQRC

SQRC is a new QR code with data reading restriction. Conventional QR code, which can be read commonly with a cell phone, has come in to use for various purposes. However, when a user wants to limit the data reading, complicated process is required such as data encryption before printing and data decryption after reading.

SQRC is a newly developed 2-D code to solve such problems. It makes it easy to encode and use non-public information including personal information and in-house information by printing QR code with SQRC compatible printer marker and reading it with a special scanner. With the development of SQRC, 2-D code is expected to be applied to new areas.

The secure QR Codes (SQRC) are the next level QR code with additional security features of segregation of private and public data. Both the codes are lookalike and have similar features but in SQRC some of the Preferred data (known as private data) can only be scanned and read by specially nominated scanners; whereas the public data can be read and encrypted by normal QR code readable scanners and mobile phones. The application of this code is tremendous in protecting every aspect of business and financial secrecy. It is one of the unique instruments in combating counterfeit and deserves the application where high level security is a demand.

SCRC is used on patient wrist bands in a hospital. When the patient scans the code on their wrist band with their smart phone they are directed to an internal web site specific to that patient where he might enter his dinner order. When the staff scans that same code with their internal devices they are directed to that patients medical records. The private data within the SQRC can be used to trigger any event as long as the scanning device knows the password.

VII. RESULT

- Integration of QR code resolve the capacity problem and store the max data
- Integrate the two QR code first QR work as a cover image and another will be original data that is encrypted
- Find the QR image that can be use in magzine,paper or every place
- The QR image can be easily scan and find the data
- In integration of QR code using stegnography use not only embedding the text but also imbed the like; audio clip, word file, video clip etc.other file
- In the stegnography use the embedding phase. In imbedding phase firstly generate the QR code and now we can use the cover image as QR code and now imbed the data in cover image that is QR code now encrypt and send to the receiver side.
- Then use the stegno decryption and remove the QR image and find the original data. The QR code can access by scanning with the help of smart phone using the android application that capable for encoding the QR image.
- These methodology uses for embedding the data, signature, embeds the very personal document and send by the mail using internet.

VIII. CONCLUSION

Encrypted QR codes are QR codes that not everyone can scan and access. They are not very common, since most QR codes are used in marketing, and the developers of those codes want them to be accessible by everyone. Secure QR (SQRC) can be made that make the scanner enter a password to be able to access the content. This is a good idea to make for employees of a company. The company can make secure QR codes that the employee has to enter the company password to view. This means people outside of the company cannot see decode the QR code without the password.

REFERENCES

[1] "QR Code, Wikipedia", http://en.wikipedia.org/wiki/QR_code
[2] "ZXING- QR Code Library ", http://code.google.com/p/zxing/
[3] "2D Code", http://www.denso-wave.com/qrcode/index-e.html
[4] "Malicious QR Code", http://www.abc.net.au/technology/articles/2011/06/08/ 3238443.htm
[5] "QR Code", http://www.tecit. Com/en/support/knowbase/symbologies/qrcode/ Defa ult.aspx
[6] "Reed-Solomon and Bose-Chaudhuri-Hocquenghem Code", http://www.berndfriedrichs.de/downloads_ecc/ ecc2010_ch08.pdf
[7] http://www.denso-wave.com/en/adcd /product/software/sqrc/sqrc.html
[8] http://vitreoqr.com/qr-code-Security-SQRC.php