# 2020 International Conference on E-Commerce and Internet Technology (ECIT)

Anti-fake Technology of Commodity by Using QR Code

Yicheng Zhan

Kings College London, London, United Kingdom [angela@cas-harbour.org](mailto:angela@cas-harbour.org)

***Abstract***—**QRcodes are often used in commodities trade, one of its applications is covering credible authentication information to avoid customer buying forged goods. This paper presents an applicable scheme which proposes an idea on how the QRcode data interacts with a customer and the interaction between qrcodes and the server. RSA and DES were used to encrypt qrcode to ensure the security of data. Besides, a scheme which enables decrypting qrcode and storing keys is presented and textbooks are served as client-server in this research. By comparing the effect to the security of qrcode when using DES or RSA only, it shows that in this way the cost of forging goods is significantly increased, which makes the rapid anti-fake identification possible.**

***Keywords-QR code; cryptography; RSA***

I. INTRODUCTION

The appearance of QRcode perfectly solved the requirement of containing more information than bar code and can be compacted conveniently. It is widely used in various areas like mobile payment, website exploring and anti-fake.

The faking commodities is a stubborn question, not only in retail industry but also in machinery manufacturing industry. Lawbreakers get the valid information of copyrighted product by using illegal approach and produce defective product to the costumers, which causes unignorable losses to both customers and the manufacturers.

Some manufacturers will make QRcodes irreproducible by using special coating to print them. Considering the QRcode has already been applied in anti-counterfeiting industry, this paper mainly discusses the way to encrypt the data of QRcode, but not the QRcode itself.

This paper proposed a doable way to encrypt QRcode, RSA and DES were used to encrypt the plaintext. With the help of mobile app and the enterprise database, customers just need to scan the QRcode and the app transfers data to the database and the database will return the feedback. Then customers can get a series of genuine information of the product. Except simply increasing the difficulty of decrypting, this paper also compares the influence of the encrypt/decrypt efficiency while using different encrypting way.

". QR CODE and ANTI-FAKE

1. *The introduction to QRcode*

Quick Response Code is a kind of two-dimensional matrix symbol, which is developed based on the bar code. One- dimensional bar code records information in the horizontal direction, but it can only represent numbers and letters. Besides,

it can not be used in mobile app due to its small storage capacity [1].



Position Detection Patterns Quiet Zone

Timing Patterns

Version Information Format Information Alignment Patterns

Data and Error Correction Code



Figure 1. An example of QRcode

As shown in figure 1, QRcode consists of position detection patterns(large square), alignment pattern(small square), two timing patterns(the line with alternating black and white cubes), the area for format and version information, the area for storage data and a blank quiet zone around it.

The minimum size for QRcode is 21x21 modules, while Data Matrix has a much more space-efficient minimum size of 10x10 modules[2]. Resulting in the Reed-Solomon error correction, the data will still be able to be read accurately even if a part of QRcode is damaged. QRcodes have four error correcting levels, L(7%), M(15%), Q(25%), H(30%) and the higher level means the area of damaged allowed is larger.

1. *QRcode Reading and anti-fake*

In a standard QR Code, corners are marked and estimated so that the inside-code can be scanned[3]. The object with different colors will reflect visible light with different wavelength. While camera is scanning QRcode, mobile phone will use image binarization to process the QRcode. Then it will do the dilate operation and obtains the outline of QRcode through the dilated image.

A gray-scale value calculation formula is used to get a standard binarization image.

After all these work, mobile phone will do the grid sampling to pixels on each node and determine whether it is “1” (black) or “0” (white). This will produce an original binary value image of QRcode, then do the data correction and decoding progress. The raw data will be transferred to actual data according to the logic decoding rules.

978-1-7281-5902-7/20/$31.00 ©2020 IEEE 1

DOI 10.1109/ECIT50008.2020.00008

QRcode itself can not be anti-counterfeiting, it works like the following. The manufacturers assign an unique QRcode to every product, then the customers use mobile phone to scan the QRcode. The verification will be done by computer and the result will be returned to the customers. The specific scheme will be discussed in part IV.

რ. RSA and DES

*1) Encryption and decryption*

Iterating

*A. Overview of RSA*

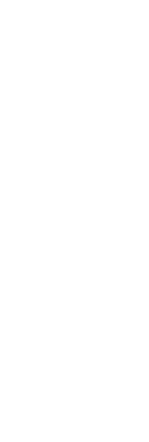
RSA is an asymmetric-key encryption algorithm. It is proposed in 1978 by professor R. I. Rivest, AShamir and M. Adleman in MIT. RSA is an exponent function based on large prime integer factoring, which is considered to be a trap-door one-way function. It is called “trap door” because the inverse function is easy to compute once a certain “trap door” information is known. The reason that they are called “one way” is because they are easy to compute in one direction but very difficult to compute in another[4].

The plaintext and cipher text are considered to be integers from 0 to n-1 in RSA(Usually the size of n is 1024).

1. *Generating pri-pub key pair*
2. First two prime integer p and q(each 1024 bits) are selected, which q  p .

Round 1

Round16



48

32

32

56

32

32

48

32

32

32

32

… … … … … … … …

… … … … … … … … …

… …



1. Compute the modulus n

#  q \* p .

1. Compute

n` 

p  1

\* q  1 and find d (private

exponent) which makes

gcd d, n`  1 .

1. Compute e (public exponent) by calculating

e \* d  1mod p  1 \* q  1 ,

which is multiplicative inverse of d[4].

1. Hence the public key is n , e and the private key is n , d .
2. *Encryption and Decryption*

To encrypt the plaintext, firstly the plaintext is represented as positive integer M. Then public key n , e  is used to calculate,

Figure 2. The structure of DES

As shown in figure 2, In the structure of iterating, every round DES will use different sub key and all sub keys *ki* are elicited from the main key *k* . DES uses Feistel network, which means the process it encrypt/decrypt data is almost the same.

Decrypting data only requires a inversed key transformation,

which is easy to implement on both hardware and software.

*E* ( *M* )  *M e* ( *m odn* ) and the sender will send receiver.

*E* ( *M* ) to

DES can be applied very efficient on 64-bit processor[5].

K. QR CODE WORKING SCHEME

Decrypting cipher text is similar to encryption. Receiver

1. *Encrypting and Generating*

uses his/her own private key n , d to calculate,

*D* ( *E* ( *M* ))  *E* ( *M* ) *d* ( *m odn* )  *M*

the plaintext from integer M.

1. *Overview of DES*

.Then he/she can extract

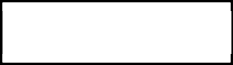
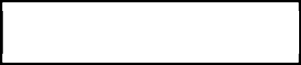
DES is a symmetric encryption algorithm, which uses 56 bits key(8 parity bits) to encrypt 64 bits grouping data. It is also an iterating algorithm, every encryption includes 16 identical

rounds and an initial and final permutation ( *IP* / *IP* 1 ). *IP* and

*IP* 1 are inverses.

2

Plain text DES key



Serialnumber: xxx-xxx-xxx-xxx sad5XxugxDqWFI=

Use DES to encrypt

Use RSA to encrypt

Figure 4. QRcode data (Example)



Generating the anti-fake label (data)

Generating a random integer between 100- 999,as the swift value to rearrange the data.

Printing the QRcode onto the product.

Use DES to encrypt the plaintext.

Generating QRcode using the encrypted data and DES key.

Use RSA to encrypt the key

key of DES.



Figure 3. Encrypting scheme

As shown in figure 3, the data will firstly be rearranged. This paper used PRNG to generate a integer between 100-999, then swift the original data according to the character sheet. The main purpose is to avoid the chosen-plaintext attack. Although the anti-faking label of most of the product is usually an unique serial number, there will still have many same substrings between different serial numbers. This attack allowed us to recover the full key in less than two hours with eight FPGAs (tests were carried out on VIRTEX1000 bg560-4) used in parallel[6]. The swift value will be written after the serial number and they will be encrypted altogether later.

Before going to the next part, both DES and RSA are already implemented by programmer in different computer languages and easy to get by downloading corresponding packages. JAVA is used to be the encryption engine.

DES is used to encrypt the plaintext and the cipher text will be the front part of the QRcode data. Since the DES is no longer secure, in order to make sure that the cipher text is unbreakable during product’s service period (the large machineries like mechanical arm and TBM have service period form a month to several years). The key will be encrypted again by RSA and then it will be the latter part of QRcode data.

Combining the cipher text and encrypted key together (shown in figure 4). QRcodes will be generated and printed onto the cover of products. DES key, private key and public key are all automatically generated by computer. Figure 5 shows a example of QRcode.

1. *Key storage*

In reality, the company will store the keys in their own database for online verification. Like any information system, a database system must run on a clean operating system and trustworthy hardware, and it must be protected against attacks over the network[7]. Here an assumption is made that when scanned data is submitted through browser the data should remain confidential on its way to the web server, the application server, and the backend DB server[8] and a secure connection between any user and the database is established[7].



Figure 5. Generated QRcode(Example)

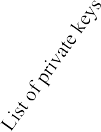
Textbook is used to simulate the database in a company. The encrypted DES keys, private key and public key will be stored in different textbooks once the QRcode is generated. These keys will be used when decrypting data.

1. *Decrypting*

As shown in figure 6, when customer scans the QRcode, the data will be uploaded to the server. Then computer will start to try to decrypt the DES key using a list of private keys from the database. If success, the DES key will be submitted to the database to match with a list of DES keys. Once the key is matched, the computer will start to decrypt the cipher text. The decrypted data will be matched with a series of serial numbers and the result will be returned to customer’s application.

3

Customer



Scan QRcode

data

List of DES keys

DecryptDES key

Matching DES key

RSA decryption success?

DES key matched?

No

**Failed**

Yes

No

Yes

**Failed**

Decrypt cipher text

Return the result

Yes

DES decryption success?

No

**Failed**

Database



Figure 6. Decrypting scheme

(**Failed** means the product is fake)

1. *Testing Efficiency*

The paper designed an experiment to compare the efficiency of proposed scheme, which is generating one thousand QRcodes in a row and each time the serial number is different. The encryption performance of the proposed scheme is compared with RSA and DES and all the situations were done by JAVA(version 12.0.1) in a computer of Intel i7 8750H CPU of 4.1Ghz and 32GB RAM 2400mhz. Each situation is performed 20 times and computes the average.

As shown in figure 7, encrypting the data with RSA takes 3 times more than using DES and 2 times more than using proposed scheme. The scheme encrypts data effectively.

|  |  |  |  |
| --- | --- | --- | --- |
| Method | QRcode size | Key Length | Encryption time(s)(avg) |
| 1.RSA | 29kb | 1024bits | 69.267 |
| 2.DES | 27kb | 56bits | 22.452 |
| 3.Scheme | 48kb | 1024/56bits | 39.484 |

Figure 7. Generating 1000 QRcodes

Although using DES takes the shortest time, but the security of data can not be guaranteed. The proposed scheme`s encryption time is 43% faster than using RSA and the data is under the same security requirement. The only difference is that the size of QRcode increased, since an additional encrypted key is after the cipher text, but the size is still within the acceptable range. The efficiency of decryption is not discussed. Since decryption happens on the side of customer, usually only several QRcodes (most of the time only one) will be submitted for verification. The typical home computer can done the decryption in few seconds or even less.

V. CONCLUSION

In this paper, a scheme of QRcode generating, scanning and verifying is proposed. The customer can verify the QRcode conveniently and the verifying process is totally automatic. A QRcode encryption method for commodity anti-faking is also proposed. The proposed method achieves strong encrypting protection and is more efficient than the classic encrypting method like RSA.

But since the situations under experiment like network connection, the security of database are ideal so if the secure of database can not be guaranteed, the encryption will be meaningless.



Figure 8. overcomplex QRcode(Example)

Also, the paper only considered the case of text encryption, QRcode could also be used for encrypting sound and image. The file size of QRcode increases significantly when the length of plaintext increases, the efficiency of encoding will be reduced and the generated QRcode will be too complex to be scanned. (As shown in figure 8) The maximum length of plaintext that the proposed scheme can encrypt is around 260 chars.

The following research will focus on optimizing the content of Qrcode, building the scanning application and the database. QRcode anti-faking will be further studied in the future work.

REFERENCES

1. T. Sun, D. Zhou, 2011 IEEE International Conference on Automation and Logistics (ICAL)-Automatic identification technology-Application of two-dimensional code, 164-168, 2011.
2. T. Falas, H. Kashani, “Two-Dimensional Bar-Code Decoding with Camera- Equipped Mobile Phones,” Fifth Annual IEEE International Conference on Pervasive Computing and Communications- Workshops (PerCom Workshops 2007), 19-23 March, White Plains, New York, USA, IEEE, 2007.
3. J. Rouillard, “Contextual QR Codes,” Computing in the Global Information Technology, The Third International Multi-Conference on IEEE, 2008.
4. R. L. Rivest, A. Shamir, L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” Communications of the Acm, 21(2):120-126, 1978.
5. E. Biham, “A Fast New DES Implementation in Software,” Lecture Notes in Computer Science, 1267, 1999.
6. G. Rouvroy, F. X. Standaert, J. J. Quisquater, et al. “Efficient uses of FPGAs for implementations of DES and its experimental linear cryptanalysis,” IEEE Transactions on Computers, 52(4):473-482, 2003.
7. U. M. Maurer, “The Role of Cryptography in Database Security,” Proceedings of the ACM SIGMOD International Conference on Management of Data, Paris, France, June 13-18, ACM, 2004.
8. J. He, M. Wang, “Cryptography and relational database management systems,” Proceedings 2001 International Database Engineering and Applications Symposium, IEEE, 2002.

4