International Journal of Engineering Research and General Science Volume 3, Issue 1, January-February, 2015
ISSN 2091-2730

QR code: A safe and secure method of authenticating legal documents

Nikita Gupta, Nagesh Mokashe, Mangesh Parihar

Jayawantrao Sawant College Of Engineering

Pune, Maharashtra, India

nikitagupta2015@gmail.com

+91-8983409849

Abstract*:* In today's world security of data is an important issue. Nowadays the forgery of legal documents is increasing at an alarming rate. To solve this problem, we provide an innovative way to authenticate the legal document (mark sheet). In this paper, we provide a method where each legal document (mark sheet) will come with a QR code. Each QR code will contain the unique details of the document which will be encoded using AES algorithm. So, if anyone tries to change the data he cannot do it because the encryption key is not known to him. The data can be retrieved using unique key. Paper proposes the idea of detecting the tampered data in legal document.

Keywords: AES algorithm, Encryption, Decryption, Quick Response (QR) code.

I. INTRODUCTION

In today's data sensitive world, it is almost impossible to secure legal data. Data of the legal documents like mark sheets, license, ATM card can be easily forged and used for greedy purposes. Example, tampering of marks or changing names in any document. Such fake mark sheet can be submitted to any company to get job or in any institution to get admission. Such forged documents can cause huge financial loss to the system. In present method the data authenticity [1] is done by humans and there are chances that it is not completely accurate. There is no method to detect such forged documents. So there's a need to secure such hard bounded legal data. In this paper, we are representing the method to detect such forged documents. In our system, we will print each document with a QR code [1][7]. This QR code will contain unique details about the document holder. These details can be encrypted using AES algorithm [8][9] and will be converted into QR code. So, whenever any person wants to check if this document is forged or not then that person can scan that QR code [1][7]. This scanned image will be sent to server. This server will decrypt this code using unique key and check for corresponding entry in database. If any record matches with it, then that record will be sent to that person.

QR Code is a type of 2D matrix barcode [1][7], which is popular because of its large storage capacity. The main feature of QR code which distinguishes it from other similar codes is that it can be easily integrated with mobile devices. In our mark-sheet system, we encode the necessary data of each student in the QR Code, like the student's name, roll number, permanent registration number (PRN), semester and year of study, and marks obtained in different subjects. All the data is encrypted using AES encryption algorithm [8][9] and stored in the QR code, and then the QR Code is printed on each mark-sheet.

II. METHOD USED

In our paper, we are using the AES algorithm. AES is based on rijndael cipher block .It is developed by two Belgian cryptographers *Joan Daemen and Vincent Rijmen*. Rijndael is a type of cipher with different block and key sizes. The block size is 128 bit and there are 3 different key sizes available i.e. 128 bit, 192 bit & 256 bit. In AES size of key and plain text need to be selected independently .The size of key and plain text decide the number of rounds to be executed. There are minimum of 10 rounds for 128 bit key and maximum of 14 rounds for 256 bit key. It is also called as symmetric key algorithm, which means same key is used for encrypting and decrypting of data. The process of converting plain text to cipher text is given below:

ALGORITHM:

(1) Do the following one time initialization process:
   (a) Expand the 16 byte key to get the actual key block to be used.
   (b) Do one time initialization of the 16 byte plain text block (STATE).
   (c) XOR the state with the key block.
(2) For each round, do the following:
   (a) Apply S-BOX to the each of the plain text bytes.
   (b) Rotate the row K of the plain text block by K bytes.

(c) Perform the mix column operation.
(d) XOR the state with the key block.

AES algorithm [8][9] process is divided into two parts:-

1) One time initialization process: This process is divided into two parts such as initialization of key and initialization of text.

a) Expand the 16 byte key to get the actual key block to be used:- In this step, we expand the 16 byte key into 11 array of each of 4*4 matrix i.e.16 byte is expanded to 11*4*4= 176 byte.

Out of 11 arrays, we use 1st array for initialization and remaining 10 arrays used for 10 rounds, 1 array per round. A word means four bytes. Therefore, our 16-byte initial key will be expanded into 176 byte key i.e. 176/4 words i.e. 44 words.

ALGORITHM FOR KEY EXPANSION:-

```
KeyExpansion (Byte key [16], word w [44])
{
  Word temp;
  for (i=0; i<4; i++)
  w[i]=(key[4*i], key[4*i+1], key[4*i+2], key[4*i+3]);
  for (i=4; i<44; i++)
  {
   temp=w[i-1];
   If(i mod 4 = 0)
   {
     temp= subWord(RotWord(temp)) XOR const[i/4];
   }
    w[i]=w[i-4]  XOR temp;
}
}
```

In first step, the original 16 byte key is copied into first four words of the expanded key. After filling the first array (for words numbered W[0] to W[3]) of the expanded key block. The remaining 10 arrays are filled one-by-one. Every added key block depend on the immediately preceding block and the block 4 positions earlier to it. i.e. W[i] depends on W[i-1] and W[i-4].

For filling four words at a time, the following logic is used:–

I) If the word W in the array is a not a multiple of 4 then simply XOR is used.

W[i] = W[i-1] XOR W[i-4]

Example 1- W [5] can be calculated by using W[4] XORing with W[1]

i.e. W[5]= W[4] XOR W[1].

II) Else perform substitution, rotate and constant operation for the value of temp.

i.e. temp = substitute(Rotate(temp)) XOR (Constant [i/4]).

Where, temp is a temporary variable used to store value of W[i-1].

Rotate – perform circular left shift on the content of the word by 1 byte.

Example - input : {00,01,02,03} will become {01,02,03,00}

Substitute – perform a byte substitution on each byte of the input word using S-BOX[8].

Constant - XOR the output of above step with constant. A Constant is a word consisting of 4 byte. The value of constant depends on round number.

b) One time initialization of 16 byte plain text block:- 16 byte plain text block is copied into a 2D 4*4 array called state. Order of copying is column wise order. First four byte of plain text is copied to first column of state array. Next four byte of plain text is copied to second column of array and so on.

c) XOR state and key:- First 16 byte of expanded key is XORed into the 16-byte state array. And the result is saved in the state array.

2) Process in each round :

The following process is executed 10 times, one per round.

a) Apply S-BOX to each of the plain text byte:- The content of state array is looked up into the S-BOX[8]. Byte-by-byte substitution is done, to replace the content of state array with the respective entry in the S-BOX [8].

b) Rotate row K of the plain text block by k bytes:- Each of the four rows of the state array is rotated to left. Row 0 is rotated by 0 byte, row 1 by one byte, row 2 by two byte and row 3 by 3 bytes.

c) Perform a Mix Columns operation: This step consists of two operations.

i) Matrix multiplication

ii) GALOIS field multiplication

i) Matrix multiplication- Each value in the column is multiplied against every value of the matrix. The results of these multiplications are XORed together to produce only 4 resulting bytes for the next state. The multiplication is performed one matrix row at a time against each value of the state column.

Example 2 -

2	3	1	1	B1	B5	B9	B13
1	2	3	1	B2	B6	B10	B14
1	1	2	3	B3	B7	B11	B15
3	1	1	2	B4	B8	B12	B16

Therefore,

b1= (b1*2) XOR (b2*3) XOR (b3*1) XOR (b4*1)

b2= (b1*1) XOR (b2*2) XOR (b3*3) XOR (b4*1) and so on.

Perform the same multiplication for all the 16 values.

ii) Galois Multiplication – the result of multiplication is actually the output of lookup of L-table, followed by the addition of result, followed by lookup of E-table. The addition means the mathematical addition, not a bit-wise AND operation. All numbers being multiplied using the MixColumn function converted to HEX will form a maximum of two digit HEX number. We use first digit in the number on the vertical index & second on horizontal index. If the value being multiplied composed of only one digit we use 0 on the vertical index.

Example 3:- If the two hex values being multiplied are AF * 8 we first look-up L-index which return B7 and then Look-up L(08) which returns 4B. Once the L-Table Look-up is Complete, we can simply add the numbers together. If addition exceeds FF value then subtract result from respective result. i.e.

B7+4B

```
   1011    0111
+  0100    1011
-------------------------------------------------
1  0000    0010   = 102 > FF  therefore, subtract FF from result

        0000  0010
+       0000  0001  (2's compliment of  FF)
```

0000 0011 = 03

- The last step is to Lookup the result in E table. Again we take the first digit to look-up the vertical index and second digit to look-up the horizontal index. i.e. E(03) = 0F

d) XOR the state with the key block : this step XOR the key for this round into the state array.

For decryption, the process can be executed in the reverse order.

III. GENERATION OF QR CODE

To generate a QR code [1][7], we first make the string of data bits. String includes the data which need to be encoded in the QR code. QR uses Reed-Solomon error detection [10] technique to generate the error correction code word for the QR code.

The resultant data is used to generate the QR code of 8 types. Each QR code have different mask pattern. Each mask pattern changes the bits according to their coordinates in the QR matrix. Mask pattern helps to make the QR code easier to read for a QR scanner. If character length exceeds 1264 characters then the same process is repeated till the time entire message is not encrypted.

CONCLUSION

In today's world as the usage of data is increasing, so is the ways to forge data. Authenticity of data is a very important issue nowadays. This paper presents an innovative method to detect such forgery of data and ensure the authenticity of data. We present a method to encrypt the data using AES and store the encrypted data in QR code. By using simple android mobile application we can generate the query to receive the original data from the server side. And finally, the data can be compared with the original data.

REFERENCES:

[1] Confidential Encrypted Data Hiding and Retrieval Using QR Authentication System: Somdip dey, Asoke nath, Shalabh Agarwal: 2013 International Conference on Communication Systems and Network Technologies, DOI 10.1109/CSNT.2013.112

[2] Symmetric key cryptosystem using combined cryptographic algorithms - Generalized modified Vernam Cipher method, MSA method and NJJSAA method: TTJSA algorithm " Proceedings of Information and Communication Technologies (WICT), 2011 " held at Mumbai, 11th – 14th Dec, 2011, Pages:1175-1180.

[3] Symmetric Key Cryptography using Random Key generator: Asoke Nath, Saima Ghosh, Meheboob Alam Mallik: "Proceedings of International conference on security and management(SAM'10" held at Las Vegas, USA Jull 12-15, 2010), P-Vol-2, 239-244(2010).

[4] New Symmetric key Cryptographic algorithm using combined bit manipulation and MSA encryption algorithm: NJJSAA symmetric key algorithm: Neeraj Khanna,Joel James,Joyshree Nath, Sayantan Chakraborty, Amlan Chakrabarti and Asoke Nath : Proceedings of IEEE CSNT-2011 held at SMVDU(Jammu) 03-06 June 2011, Page 125-130(2011).

[5]Cryptography and Network Security, William Stallings, Prentice Hall of India.

[6] Cryptography & Network Security, Behrouz A. Forouzan, Tata McGraw Hill Book Company

[7] "QR Code, Wikipedia", http://en.wikipedia.org/wiki/QR_code [Online] [Retrieved 2012-02-09]

[8]Cryptography and Network Security, Atul Kahate,Tata Mcgraw-Hill Education

[9] "AES algorithm, wikipedia" http://en.wikipedia.org/wiki/Advanced_Encryption_Standard[online].

[10] Reed and G. Solomon, "Polynomial codes over certain finite fields", Journal of the Society for Industrial and Applied Mathematics, 8(2):300–304, 1960