Data Encryption Algorithm using Asymmetric Key Derived from Fingerprint Biometric Features

Prof. Dr. Tarik Zayed Ismael1, Ahmed Saad Names2
University of Baghdad, Electrical Engineering Department, Baghdad, Iraq1,2

Abstract: Security is one of the major concerns in the present because of increasing the information quantity and theft. Due to these causes, an efficient security system is more important to protect the personal information. This leads to use password, keys, etc. to secure the information. Fingerprint Encryption is preferable biometric technique for securing text because of its parallelism, vast storage and fast computing quality. In this paper, we use the features provided by the fingerprint of the information sender to extract a new encryption method depending on the distribution of the hells and valleys of the fingerprint since this distribution differs from one finger to another and from one person to another. The proposed method will be used for text data transmission. The types of encryption key generators which will be used through this research are the private key generators. Extracted the feature vector from fingerprint. Then the feature vector is converted to an encryption key. In general, encryption algorithms use this fingerprint key to encrypt and decrypt the data. In encryption and decryption, we use a new algorithm which depends on look up table to encrypt and decrypt message. The security is enhanced in this paper by using fingerprint encryption technique.

Key Words: Encryption Algorithm, Asymmetric key, Fingerprint biometric, Decryption Algorithm.

1. INTRODUCTION

Data security is very much necessary for data security of the personal computer. Used the Cryptography techniques are using for secure data. Cryptography is the ability for safe storage and transport sensitive data. Cryptography, to most persons, is concerned with preserving communications secret. Cryptography means “Hiding a information by first converting it into an unintelligible form by ciphering and then converting back to intelligible form by deciphering [1].

The technique of protecting data by converting an encrypted message into an unreadable format, called cipher text only those who possess a privet key can decipher or decrypt the message into plain text. Encrypted messages can sometimes be broken by cryptanalysis and is also called code breaking. In short, using cryptography data is first encrypted into another form and then transmitted. Thus, in Cryptography there are different steps which are as follows:

1. In First, encrypt the message.
2. In Second, step processing of message is done which contain a private key.
3. At last the encrypted message is decrypted by using private key [10].

There are two types of encryption methods: Symmetric Key or Private Key Cryptography and Asymmetric or Public Key Cryptography. In our algorithm will use a symmetric.

In general, symmetric encryption scheme has five ingredients: Plaintext, Encryption algorithm, Secret key, Cipher text and Decryption algorithm. In any encryption algorithm, there are two requirements [1]

1. We need a strong encryption algorithm and strong key.
2. Sender and receiver should have used the same key.

Fingerprint is one of the most common physical biometric patterns analyzed that is used to purposes of information security.

A biometric is a unique, measurable, biological characteristic or trait for automatically recognizing or verifying the identity of a human being. A statistical analysis these biological characteristic is known as the biometrics science [3].

In present, fingerprint technologies are used to analyze characteristics of biometrics science for security purposes. Each fingerprint contains two types of features. These features are called end ridge and bifurcation ridge.

The security is depended on the strong of the key [3]. In the traditional cryptographic algorithms, such as “advanced encryption standard” (AES), “data encryption standard” (DES) and “Rivest-Shamir-Adelman” (RSA) etc., data is encrypted using privet or public key. The short keys are easy to be remembered, but they are also easy to be broken. And the long keys are difficult to be broken, but they are also difficult to memorized and have to be stored in somewhere, which can be stolen or lost.

Biometric encryption, which use the features of biometric to encrypt data. Some biometric cryptographic algorithms have been developed based on fingerprints [7], iris [8], signature [9] and DNA [4] etc.

The fingerprint encryption offers a new mechanism for key security by using a fingerprint to secure the cryptographic key. Instead of entering a password to access the Cryptographic key, the use of this key is
guarded by fingerprint encryption. When a user wants to access a secured key, the user will be prompted to allow for the capture of a fingerprint sample. Then the key is released and can be used to encrypt or decrypt the desired data. The fingerprint encryption will offer both convenience, as the user does not need to remember a key, and secure identity confirmation, since only the real user can release the key [3].

2. FINGERPRINT IMAGE ENHANCEMENT, PREPROCESSING AND FEATURE EXTRACTION

To use the image of fingerprint in encryption algorithm must be applied many of intermediate steps. In these steps, will enhance the fingerprint image by applying some of mathematical equation to get on the enhanced image. And then will extract feature from Enhanced image [6]. All intermediate steps in respectively are.

1. Segmentation.
2. Image enhancement.
4. Region of Interest [5].
5. Thinning [2].
6. Feature extraction

Will use the Crossing Number (CN) concept to extract the feature from thinned image.

In this method, a window of 3x3 pixels is used to check each pixel neighborhood in the image and the CN value is determined as half the sum of the differences between two neighbor pixels in eight pixels which is Surrounding of this window.

The most stableridges, which called minutiae, are:
1) ending
2) bifurcation.

Where a ridge ends suddenly. This point calls ending. And where the ridge branches in to two ridges. this point calls bifurcation. CN for ridge ending equal to one and CN for ridge bifurcation equal to three [2]. Examples of minutiae are shown in Fig. 1 [6].

![Minutiae](image)

Figure 1. Minutiae. (a) A minutiae can be characterized by its position and its orientation.

3. ENCRYPTION AND DECRYPTION

A. Code generation

Step 1. Find average point of minutiae (ridge ending and ridge bifurcations), 
\[ \begin{align*}
X' &= \frac{\sum Xi}{i} \\
Y' &= \frac{\sum Yi}{i}
\end{align*} \] (1). When i is equal to number of minutiae.

Step 2. Convert that minutiae (ridge ending and ridge bifurcations) to polar form i.e. (magnitude and theta) by calculate the length (LE) and angle (\( \theta \)) between each minutiae and average point.

\[ L_{EI} = \sqrt{(Xi - X')^2 + (Yi - Y')^2} \] (2);

\[ \theta_i = \tan^{-1}\frac{Y_i - Y'}{X_i - X'} \] (3)

Step 3. Find the maximum value of length from this length
Step 4. Normalize this lengths by divided each value on the maximum value.

\[ L_{Ei} = \frac{L_{EI}}{\max(L_{EI})} \] (4).

Step 5. Write the codewhich represented the vector of (\( L_{Ei} \) norm and \( \theta \)).

Note. We can derive the fingerprint’s code from minutiae ending or minutiae bifurcation or merge between them

B. Create look up table

Create look up table \((M*N)\) where \( M \) is the number of rows which depend on the number of symbols that wont to encrypt and \( N \) is the number of columns which depend on the code i.e.(each range of length and angle of fingerprint code will refer to one column),as shown in table (1)

C. Encryption

Step 1. Select the same sequence for key and message. I.e. (first key to first letter). If the length of key is shorter than letters of message will repeat that code.

Step 2. Select the column from look up table depends on the first number of fingerprints’ keys (norm \( L_{EI} \) and \( \theta \)).

Step 3. Choose the row depend on the location of this litter at first column

Step 4. Intersection between that column and row to choose cipher letter

Step 5. Repeat that four steps above to encrypt another letter.
Table (1) show how generate look up table

<table>
<thead>
<tr>
<th>Number of row</th>
<th>Number of column</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>At ORIGINAL MESSAGE</td>
<td>LEnorm (0 - 0.7, Θ&lt;180)</td>
<td>LEnorm (0.7-1, Θ&lt;180)</td>
<td>LEnorm (0 - 0.7, Θ&gt;180)</td>
<td>LEnorm (0.7 - 1, Θ&gt;180)</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>A</td>
<td>X</td>
<td>N</td>
<td>H</td>
<td>J</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>B</td>
<td>C</td>
<td>Q</td>
<td>Y</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>C</td>
<td>F</td>
<td>W</td>
<td>U</td>
<td>K</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>D</td>
<td>G</td>
<td>E</td>
<td>C</td>
<td>Z</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>E</td>
<td>D</td>
<td>T</td>
<td>%</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>F</td>
<td>A</td>
<td>B</td>
<td>G</td>
<td>P</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>G</td>
<td>M</td>
<td>C</td>
<td>A</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>H</td>
<td>R</td>
<td>F</td>
<td>M</td>
<td>E</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>I</td>
<td>N</td>
<td>G</td>
<td>R</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>J</td>
<td>Q</td>
<td>X</td>
<td>N</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>K</td>
<td>W</td>
<td>L</td>
<td>X</td>
<td>D</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>L</td>
<td>E</td>
<td>O</td>
<td>Q</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>M</td>
<td>Z</td>
<td>P</td>
<td>W</td>
<td>R</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>N</td>
<td>2</td>
<td>*</td>
<td>S</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>O</td>
<td>T</td>
<td>D</td>
<td>E</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>P</td>
<td>#</td>
<td>A</td>
<td>T</td>
<td>Q</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Q</td>
<td>H</td>
<td>M</td>
<td>B</td>
<td>W</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>R</td>
<td>Y</td>
<td>R</td>
<td>D</td>
<td>S</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>S</td>
<td>&amp;</td>
<td>J</td>
<td>L</td>
<td>F</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>T</td>
<td>J</td>
<td>I</td>
<td>O</td>
<td>G</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>U</td>
<td>K</td>
<td>K</td>
<td>P</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>V</td>
<td>I</td>
<td>Z</td>
<td>V</td>
<td>M</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>W</td>
<td>L</td>
<td>H</td>
<td>J</td>
<td>H</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>X</td>
<td>O</td>
<td>I</td>
<td>Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>Y</td>
<td>P</td>
<td>U</td>
<td>K</td>
<td>U</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>Z</td>
<td>V</td>
<td>S</td>
<td>Z</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>space</td>
<td>U</td>
<td>Y</td>
<td>!</td>
<td>L</td>
<td></td>
</tr>
</tbody>
</table>

Figure (2): block diagram illustrates encryption steps
For example: to our encryption algorithm. Took asample of minutiae from fingerprint’s image

\[ X = [82, 125, 201, 222, 281, 6, 225, 201] \]

\[ y = [7, 9, 14, 250, 251, 252, 253, 10] \]

At first generated code by apply generated code algorithm.

Equation (1, 2, 3, 4)

And get to code as shown in table (2). And then generated lock up table (for our example contain 4 columns and 27 rows) as shown in table 1. If wont to increase encryption strength take moor rows and columns.

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>( L_{ei} )</th>
<th>( \text{Len}_i )</th>
<th>( \theta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>82</td>
<td>7</td>
<td>150.6273</td>
<td>0.7448</td>
<td>235.2417</td>
</tr>
<tr>
<td>125</td>
<td>9</td>
<td>129.0788</td>
<td>0.6382</td>
<td>250.6000</td>
</tr>
<tr>
<td>201</td>
<td>14</td>
<td>121.3583</td>
<td>0.6000</td>
<td>285.8400</td>
</tr>
<tr>
<td>222</td>
<td>250</td>
<td>130.9583</td>
<td>0.6475</td>
<td>65.5877</td>
</tr>
<tr>
<td>281</td>
<td>251</td>
<td>165.0979</td>
<td>0.8163</td>
<td>46.7487</td>
</tr>
<tr>
<td>6</td>
<td>252</td>
<td>202.2500</td>
<td>1.0000</td>
<td>143.1655</td>
</tr>
<tr>
<td>225</td>
<td>253</td>
<td>134.9382</td>
<td>0.6672</td>
<td>64.9542</td>
</tr>
<tr>
<td>201</td>
<td>10</td>
<td>125.2111</td>
<td>0.6191</td>
<td>285.3404</td>
</tr>
</tbody>
</table>

Table (2) show generated code

To encrypt message (HELLO WORLD).

Step 1. Choose the same sequence for cod and message. I.e. (first code to first letter). If the length of cod is shorter than letters of message will repeat that code.

Step 2. Depend on the fingerprint’s code choose the column from table (1).

Step 3. In this column will find a location of the symbol that wrote in the encrypted message.

Step 4. Depending on the rows’ location of that symbol will choose the letters from the first column.

Step 5. Repeat three steps above to decrypt all messages.

To decrypt (E%QEDYLESQC) at first depend on code vector will choose 5th column and find the row of letter E which located in 8th row and then take the 8th letter from first column which refer to (H).

And then take second code and second letter and so on to decrypt all messages and get to original message (HELLO WORLD).

4. Security strength

An attack is a most threat to any security system. For a fingerprint encryption system, not only the encryption information, e.g. the password, but also the fingerprint data must be protected against attacks. In this section, will list some of typical attacks and analyze how the proposed protocol defends against

A. Biometric template attack

In this algorithm, Key generates from fingerprint feature, i.e. length and angle between average point and minutiae. It is noticeable that one fingerprint can only contain limited number of minutia points, approximately between 20—40. If attackers acquire minutia information successfully, they can narrow down the search range of Key. In other words, the cryptographic key space created by 20—40 minutiae points is small [7].

The proposed solution to solve this problem by increasing the total number of minutiae by add noise through image enhancement and this noise will increasing the number of minutiae and also will increase the key strength.
B. Brute force attack

Involves trying every possible key until an intelligible translation of the cipher text into plaintext is obtained. On average, half of all possible keys must be tried to achieve success [1]. The security strength of an algorithm cannot exceed its key length. In our protocol, Key is generated from a set of minutia points and number of rows and columns in table.

To break a key by the brute force attack for a given key length

\[
\text{Time needed to break the fingerprint key in second } = \frac{N_{C}^{L_{k}}}{2 \times 10^{6}}\]

\(N_{C}\)=number of column in table.
\(L_{k}\)=length of fingerprint key.
Divided by 2, for averaging
Divided by1000000, assuming that it takes 1\(\mu\)S to perform a single decryption, [1]
To calculate the time that need to break a table, which used for encryption, will use equation (6)
Time need to break a table=\(N_{C}^{*}(N_{R}!)\) (6)
When \(N_{R}\) is number of rows i.e. (number of symbol that use in encryption algorithm).
\(N_{C}\) is the number of columns.

5. EXPERIMENTAL RESULTS AND ANALYSIS

A. Encrypted message by fingerprint code and using Mat lab environment

- At first derive the fingerprints code from minutiae that represent ridge ending to encrypt this message (HELLO WORLD). And get to this cipher (R3&T@M9@X91)
- Then derive the fingerprints code from minutiae that represent ridge bifurcation to encrypt this message (HELLO WORLD). And get to this cipher (VDX#F7YYX?)
- At last derive the fingerprints code by using combination of two minutiae (ridge bifurcation and ridge ending) to encrypt this message (HELLO WORLD). And get to this cipher (G9XLF7YYXBP)

B. Comparison between fingerprint encryption and AES algorithm

AES proves to be a better security algorithm than DES and Triple DES [11], due to this reason will compare the performance of our algorithm with AES.

Shown in figure (4) the Comparison between fingerprint encryption and AES algorithm:

At first, test performance of AES by compute half of all possible keys divided by (1000000) in figure (4a). Then Test the performance and measure security against brute force attack of our algorithm and get to the result that shown in figure (4 b).

By observing the results in Figure 3 we compare between our algorithm and AES algorithm.

1-The security of AES algorithm depended on a key length while in our method the security depended on two parameters (key length and number of rows in table).
2-our method takes much more time to break by the brute force attack for a given key length.

![Figure (4); the Comparison between fingerprint encryption and AES algorithm. (a) AES algorithm (b) Fingerprint encryption](image)

C. Speed of fingerprint encryption

The speed of encryption system is measured by size of encrypted data, in kilo byte, peer one second.

The result of this test shown in figure (5)

![Figure (5). Fingerprint encryption speed](image)
6. CONCLUSIONS

This paper proposed a novel fingerprint cryptosystem. The system extracted vector feature from fingerprint and used the modified algorithm to encrypt and decrypt messages. Experimental results show that the proposed fingerprint cryptosystem can work effectively, and observe that can encrypt message using three fingerprint’s code derive from minutiae (ending, bifurcation and merge between them) and get to three different message. The result shows that our algorithm is very strong and it is almost impossible to be cracked.

REFERENCES