Journal of Information Hiding and Multimedia Signal Processing ©2018 ISSN 2073-4212
Ubiquitous International Volume **9**, Number **1**, January **2018**

Email Encryption System Based on AES Algorithm and DH Algorithm

Jia Liu, Chunlei Fan, Xingyu Tian and Qun Ding

Electronic Engineering College
Heilongjiang University
Harbin, China
18846089521@163.com, qunding@aliyun.com

Received January, 2017; revised July, 2017

Abstract. *Email has become an important application in the developed world, it is familiar to most of us in our daily life. However, many different emails transmit information through plaintext, it may brings about information security problems. So, it is necessary to design a email encryption software that provides email encryption and decryption. Email encryption system is presented in this paper, which resides on the user's computer between the email client and the email server, intercepting, encrypting, decrypting, and authenticating email communication. This paper studied the design and implementation of AES algorithm encryption system. At the same time, because the space of the key is large and high security, we put the key exchange algorithm of Diffie-Hellman to our encryption system, which enhances anti-attack capability greatly in encryption system and guarantees security of information transmission effectively.*
Keywords: Email encryption, AES algorithm, Diffie-Hellman algorithm

1. **Introduction.** As the speed of network development is faster than data security thoughts, the security issues became more prominent today as the network is widely applied. Since a lot of sensitive information is exchanged through E-mail and the network is quite open, users need to transmit information safely and efficiently urgently [1, 2]. AES is used as a new generation data encryption standard encompasses a lot of advantages, such as, high security, high performance, flexible and easy to use. However, with the development of Internet, there is increasingly unsafe factors and high demand of operation efficiency. Although AES key expansion algorithm has direct and efficient characteristics, there is a defect that all keys can be cracked by means of any round sub key [3]. There is three improved algorithms for the defect, operation efficiency of the first and the second algorithms is reduced, the ability of third round and before to defend an attack is weak, although the third algorithm has same attack power and brute force after two rounds key expansion, since algorithm structure is too complex, it is hard to realize [4]. S-box has a number of defects, such as, the period of affine transformation pair is short, the iterative output cycle is short and the algebraic expression has only 9 items [5]. The new S-box is constructed by affine transformation of the byte and then inverse the element, and an affine transformation is made to overcome the defect of the original S-box, the new has better algebraic properties [6]. By changing the structure of AES algorithm, there are reusable units in encryption and decryption, and a reconfigurable design method is proposed, which is very suitable for hardware implementation [7, 8]. It is proposed that the four transformations in the AES round transformation can be simplified into one step

by using the look-up table to improve the efficiency of the AES algorithm. However, this document does not solve the hidden dangers and the problem of the key expansion in decryption [9]. Diffie-Hellman key exchange algorithm can let sender and recipient both in the same public network transmission of sensitive data. Both sides of the transmission hold a public key and a private key, the two parties share a session key, and then transmit the sensitive data. In this way other people do not know the session key, the security of sensitive data can be guaranteed [10]. A very important aspect of the security analysis of DH cryptosystems is the security and integrity of its single or partial bits. Or a lower bound on the number of polynomials or weights when polynomials are used to recover the negotiated key. 32bits from the user's public key are equally difficult to compute the entire negotiation key [11], Vehraul proves that the DH protocol key is computed in the extended domain of the finite field, information is easy to generalize to any extended domain cryptosystem [12].

2. AES algorithm.

2.1. AES Key Expansion Analysis and Improvement.

Based on the table look-up method, AES key expansion process shown in Figure 1. Each round of operation depends on the previous round, followed by pushing down to get the desired arbitrary round key. This kind of key expansion method has the advantage of high efficiency and immediacy, but if the attacker gets one round key, the whole key can be cracked. Because each word w_i is related to w_{i-1} and w_{i-4}, in other words, if any two of them are known, we will get the third. Assuming that the attacker knows one of the keys w_i, w_{i+1}, w_{i+2}, and w_{i+3} of AES, w_{i-1} can be deduced by w_{i+2} and w_{i+3}, w_{i+1} and w_{i+2}, w_{i-2}, w_{i-3} is obtained by w_i and w_{i-1}, and w_{i-4} is obtained by w_{i-1} and w_i. Thus, all sub keys of the previous round are obtained, and similar method is used to obtain the next round of key, so we can get all the keys.

In view of above security risks, this paper proposes an improved key expansion algorithm from the aspects of anti-attack strength and taking into account the execution time of the program: the initial key is unchanged, the first round of expansion key is set with a set of initial key and new key to fill, on the basis of the new key, AES inherent algorithm is used for key expansion until all the sub-key is generated. The principle of this method is shown in Figure 2. After this change, since there is no relationship between the initial key and the extended key, the attacker can not deduce the entire key from a round key. If we use the exhaustive key attack, we assume that the seed key length is k bit, the best case of exhaustive key attack is 1 and the worst case is 2^k. Since the probability of each case is equal, the average complexity is

$$\sum_{i=1}^{2k} \frac{1}{2^k} \times i = \frac{1}{2^k} \times \sum_{i=1}^{2k} i = \frac{1}{2^k} \times \frac{(1+2^k) \times 2^k}{2} = \frac{1}{2} + 2^{k-1} \approx 2^{k-1}.$$

For 10 rounds of AES algorithm, the attacker needs to try 2^{127} possible keys on average, and the key expansion algorithm in this paper makes the attacker need to try 2^{255} possible keys on average. In terms of current computing power, completing this exhaustive search will take at least hundreds of millions of years. Therefore, the improved key expansion method is only made a small part of the changes in the original method, which both overcomes the original security risks and ensures the efficiency of the program.

2.2. Analysis and Optimization of Mixcloumns and Inverse Mixcloumns.

AES algorithm encryption and decryption operation time-consuming is different, the reason is that the algorithm complexity of Mixcolumns and Inverse Mixcloumns is distinct. In a Mixcolumns transformation, each column of the state is treated as a polynomial over

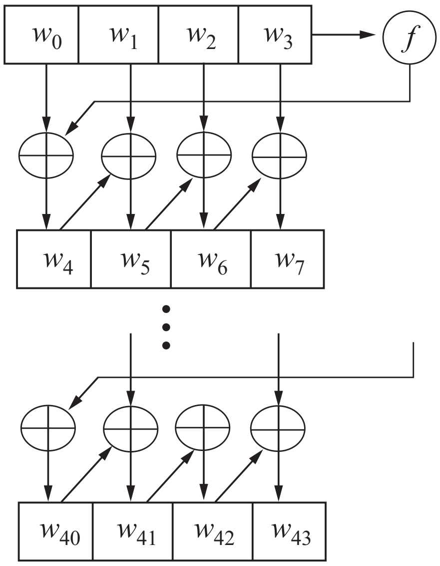

FIGURE 1. AES key expansion process

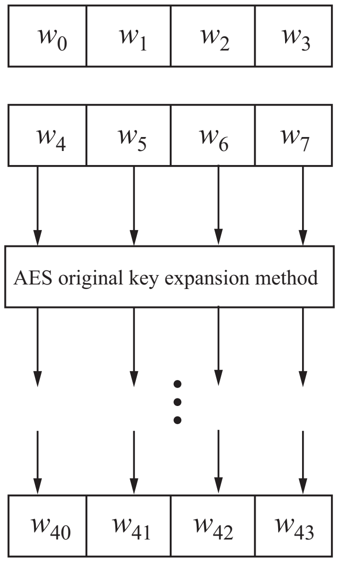

FIGURE 2. Improved key expansion algorithm

$GF(2^8)$ and associated with a fixed polynomial $C(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$, and mod the modulo polynomial $x^4 + 1$. The matrix is expressed as follows:

$$\begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \begin{pmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{pmatrix} = \begin{pmatrix} b_{00} & b_{01} & b_{02} & b_{03} \\ b_{10} & b_{11} & b_{12} & b_{13} \\ b_{20} & b_{21} & b_{22} & b_{23} \\ b_{30} & b_{31} & b_{32} & b_{33} \end{pmatrix} \tag{1}$$

Inverse Mixcloumns process can also be expressed as the matrix multiplication:

$$\begin{pmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0E & 0D & 09 & 0E \end{pmatrix} \begin{pmatrix} b_{00} & b_{01} & b_{02} & b_{03} \\ b_{10} & b_{11} & b_{12} & b_{13} \\ b_{20} & b_{21} & b_{22} & b_{23} \\ b_{30} & b_{31} & b_{32} & b_{33} \end{pmatrix} = \begin{pmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{pmatrix} \tag{2}$$

It is not difficult to see that the decryption process is much more complex than the encryption process. In the encryption process, the Mixcloumns transform needs to perform four XOR additions and two times xtime multiplications. The Inverse Mixcloumns transform in decryption process requires nine XOR additions and twelve times xtime multiplications [13]. As multiplication consumes more time and space resources, resulting in there is delay during decryption process relative to the encryption process, in practice, the process is often difficult for users to accept.

Reference [14] decomposes the decryption matrix into the product of two simpler matrices to reduce the multiplication times and improve the decryption rate. However, this method is more complex to achieve, promotion of efficiency is not obvious. In this paper, we use the theorem 1 to find Mixcolumns and Inverse Mixcloumns with the simplest form.

Theorem 2.1. *In finite field $GF(2^8)$, if there is a linear matrix A,*

$$A = \begin{pmatrix} a & b & c & d \\ d & a & b & c \\ c & d & a & b \\ b & c & d & a \end{pmatrix}, \quad a, b, c, d \in GF(2^8)/\{0\},$$

if $A^{-1} = A$*, then*

$$A = \begin{pmatrix} a & b & c & b \\ b & a & b & c \\ c & b & a & b \\ b & c & b & a \end{pmatrix},\ a^2 + c^2 = 1.$$

It is proved that g is a generator in finite field $GF(2^8)$, $\alpha, \beta, \gamma, \rho$ are orders of elements a, b, c, d respectively. The following equation can be constructed by $AA^{-1} = 1$:

$$\begin{pmatrix} a & b & c & d \\ d & a & b & c \\ c & d & a & b \\ b & c & d & a \end{pmatrix} \begin{pmatrix} a & b & c & d \\ d & a & b & c \\ c & d & a & b \\ b & c & d & a \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

We can get $\begin{cases} a^2 + c^2 = 1 \\ b^2 + d^2 = 0 \end{cases}$, which is $\begin{cases} g^{2\alpha} + g^{2\gamma} = 1 \\ g^{2\beta} + g^{2\rho} = 0 \end{cases}$, since $g^{2\beta} + g^{2\rho} = 0$, $1 \leq \beta,\ \rho \leq 225$, then $2\beta = 2\rho$ or $2\beta = 2\rho \pm 255$, but 2β is even number, $2\beta = 2\rho \pm 255$ is invalid. So $\beta = \rho$, $b = d$. Above all,

$$A = \begin{pmatrix} a & b & c & b \\ b & a & b & c \\ c & b & a & b \\ b & c & b & a \end{pmatrix},\ a^2 + c^2 = 1.$$

According to above theorem, if

$$M = \begin{pmatrix} 2 & 1 & 3 & 1 \\ 1 & 2 & 1 & 3 \\ 3 & 1 & 2 & 1 \\ 1 & 3 & 1 & 2 \end{pmatrix} = M^{-1},$$

we use this matrix to replace Mixcolumns and Inverse Mixcloumns in the original algorithm, so Mixcolumns and Inverse Mixcloumns consume the same computing resources, which solves the time delay problem of decryption relative to encryption with high practical value.

3. **Diffie-Hellman key exchange algorithm.** Diffie-Hellman key exchange algorithm is a key to ensure the safety of the security network through the algorithm, it is a key exchange protocol proposed by Whitefield and Hellman Martin in 1976. The secret of this algorithm is that both sides of the secure communication can use this method to determine the symmetric key [15]. You can use this key to encrypt and decrypt. In order to negotiate one key between two communication participants, it is necessary to make sure that the information they receive in the process of the protocol is indeed from the real people. Diffie-Hellman key exchange protocol can fully guarantee the security of the key, the main principle of the algorithm is shown in figure 3. The key exchange algorithm is very important for a lot of network security applications.

3.1. **The Security Analysis of DH Key Polynomial Transformation.** We assume that $F(X) = \sum_{i=1}^{m} c_i X^{e_i} \in F_q[X], c_1, \cdots c_m \in F_q^*, e_i \neq e_j (\bmod t), i \neq j$. Suppose that given a interrogator responder $O_{F,\varepsilon}$. Satisfying to any $x \in [0, t-1]$, and enter the value γ^x and γ^y, for at least given εt values of $F(\gamma^{xy})$, $y \in [0, t-1]$ and give the error message for the remaining y values.

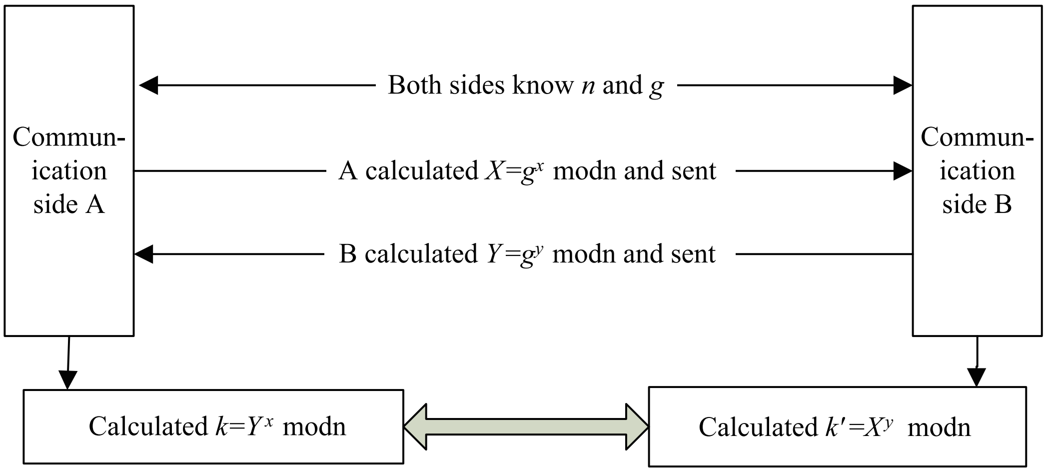

FIGURE 3. The principle of Diffie-Hellman key exchange algorithm

Theorem 3.1. *We assume that t is a prime,* $m \geq 2, 4t^{-\frac{1}{m-1}} \leq \varepsilon \leq 1$, *given an interrogator responder* $O_{F,\varepsilon}$, *then there is a probability polynomial algorithm, which for any* $(x, y) \in [0, t-1] \times [0, t-1]$, *input* γ^x *and* γ^{xy}, *output* γ^{xy}, *and just ask* $m\varepsilon^{-1}$ *times on average, only* $O(m \log q)$ *times operations among* F_q *at a time.*

Proof: If $x = 0$, for any input, there is correct output. We consider that $(x, y) \in [0, t-1] \times [0, t-1]$. Suppose that the set U is a set of questionnaires that can be derived from the input γ^x and γ^{y+u}, the output $\gamma^{x(y+u)}$ of $u \in [0, t-1]$, it is obviously that $|U| \geq \varepsilon t$. Let $\theta = \gamma^x$, suppose that k is a positive number, we have chosen $k-1$ integers $u_1, \cdots u_{k-1} \in U$ satisfies the determinant

$$\det(\theta^{e_i u_j})_{i,j=1}^{k-1} \neq 0 \tag{3}$$

Then we choose $u_k \in U$ which satisfies

$$\det(\theta^{e_i u_j})_{i,j=1}^{k} \neq 0 \tag{4}$$

If $\det(\theta^{e_i u_j})_{i,j=1}^{k} = 0$, then $\Delta_1 Tr(\theta^{e_k u_k}) + \cdots \Delta_1 Tr(\theta^{e_1 u_k}) = 0$, We assume that $\Delta_1 = \det(\theta^{e_i u_j})_{i,j=1}^{k-1} \neq 0$, the number of satisfying equations (4) is

$$|U| - 2\left(\frac{1}{1-\left(\frac{1}{q-1}\right)^{\frac{1}{m-1}}}\right)t^{1-\frac{1}{k-1}} \geq |U| - 2\left(\frac{1}{1-\left(\frac{1}{q-1}\right)^{\frac{1}{m-1}}}\right)t^{1-\frac{1}{m-1}} \approx |U| - 2t^{1-\frac{1}{m-1}}$$
$$\geq t(\varepsilon - 2t^{-\frac{1}{m-1}})$$

Then we find out the probability of satisfying equations (4) $u_k \in U$ is $\varepsilon - 2t^{-\frac{1}{m-1}}$, in actual application t is usually taken as a large prime, $u_k \in U$ can be found with the same probability finding $u_1, \cdots u_{k-1} \in U$, that is about $m\varepsilon^{-1}$ times, we can get $u_1, \cdots u_m \in U$.

Let $A_j = F(\theta^y \theta^{u_j}), j = 1, 2, \cdots m$, which satisfies $\det(\theta^{e_i u_j})_{i,j=1}^{m} \neq 0$. In fact, we obtain the following non-singular linear system of equations $\sum_{i=1}^{m} c_i \theta^{u_1 e_i} \theta^{e_i y} = A_j, j = 1, 2, \cdots m$. $(c_1\theta^{e_1 y}, \cdots c_m \theta^{e_m y})$ can be obtained, then we get $(\gamma^{e_1 xy}, \cdots \gamma^{e_m xy})$. Since $m \geq 2$, t is a prime number, there is at least one element in $e_1, \cdots, e_m$ and t are co-prime factors, $(e_1, t) = 1$, $f_1 e_1 \equiv 1 (\bmod t)$, then $\gamma^{xy} = \gamma^{e_1 xy^{f_1}}$.

The above is the main consideration which from the perspective of polynomial conversion, the security of the analysis of two parties from the public key to restore the key can be considered as DHD (decision diffie-hellman) problem of polynomial transformation analysis.

3.2. **Bit security analysis.** S. C. Pohlig and M. E. Hellmen proved that the security of discrete logarithms on $GF(p)$ depends on the large prime factor of $p-1$, so in practical cryptographic applications, $p = 2^k q + 1$ is usually used, if there is tronger requirements, we choose $p = 2q + 1$ (q is also a large prime), such a parameter selection can achieve smaller bit leakage, but if we do operation on the whole $GF(p)$, there is still one bit leakage. If you use the elements with large prime numbers q on $GF(p)$ as substrate $\gamma^x(\bmod p)$, there will be no bit leakage on the exponent x, in this case, it will be pointed out that the first bit is important, this paper point out that bit relationship of each element in general case.

Definition 3.1. *we assume that that $p = 2^k q + 1$, q is also prime number, $F_p^* = \langle\gamma\rangle$, for any $0 < x < p - 1$. For a given value of $y \equiv \gamma^x(\bmod p)$, the minimum k-bit of x can be recovered in polynomial time.*

Proof: Firstly, it is judged whether $y \equiv \gamma^x(\bmod p)$ is $\bmod p$ quadratic residue, it is judged that $y^{\frac{p-1}{2}} \equiv \gamma^{x\frac{p-1}{2}} \equiv 1(\bmod p)$, in this way can we judge the parity of x (the first bit). Suppose that x is an odd number. $x = 2x_1 + 1$, the parity of x_1 (second bit) is judged as follows. Let $y_1 \equiv \frac{y}{\gamma} \equiv \gamma^{2x_1}(\bmod p)$, then judge $y_1^{\frac{p-1}{2^2}} \equiv 1(\bmod p)$, in this way can we judge the parity of x_1.

And so on you can calculate $x = 2^k x_k + c$, where c is a constant that has been found, x_k is unknown. From the above results, it is not difficult to deduce that when $p = 2q+1$, the discrete logarithm problem with γ as base always has one leakage. The above proofs can not be performed similarly because of $(2, q) = 1$. The following further results illustrate this problem and can be generalized in parallel to a general finite cryptosystem.

Theorem 3.2. *Suppose that γ is the generator of LUC subgroup order $l = 2^s p_1^{a_1} \cdots p_t^{a_t} q$, then for any $i \in \{1, 2, \cdots t\}$ can be restored by $V_x(P)$ using p_i represent $x(1 \le x \le l-1)a_i$ bit low bit value, where q is prime, $p_1, p_2 \cdots p_t$ is less than the number of an upper bound B.*

Proof: It is obvious that there are two possible values for any $V_x(P)$ secret exponent, x and $l - x$. Firstly, we solve the following equation $X^2 - V_x(P)X + 1 = 0$. The answers are γ^x and γ^{l-x}. The low bit values of x are recovered by generalized parity detection.

We assume that $p_1 = 3$, $x = x_0 + 3x_1 + 3^2 x_2 + \cdots 3^{r-1} x_{r-1}$, $0 \le x_i \le 2, i = 0, \cdots r-1$, the method is as follows: (1) If $y \equiv (\gamma^x)^{\frac{1}{3}} \equiv 1(\bmod p)$ then $x_0 = 0$, otherwise $x_0 = 1$ or $x_0 = 2$. (2) If $y_1 \equiv (\gamma^{x-1})^{\frac{1}{3}} \equiv 1(\bmod p)$ then $x_0 = 1$, otherwise $x_0 = 2$. Thus we recover the lowest bit value of x, and

$$y_2 \equiv \gamma^{3x_1 + 3^2 x_2 + 3^{r-1} x_{r-1}}(\bmod p) \tag{5}$$

The open cubic root of equation (5) has three possible values, $x_1 + 3^1 x_2 + \cdots + 3^{r-2} x_{r-1}$, $x_1 + 3^1 x_2 + \cdots + 3^{r-2} x_{r-1} + \frac{l}{3}$, $x_1 + 3^1 x_2 + \cdots + 3^{r-2} x_{r-1} + \frac{2l}{3}$. Because $\frac{l}{3} = < \cdots \underbrace{0 \cdots 0}_{a_1 - 1} >_3$, this does not affect $x_1, x_2, \cdots$ and the $a_1 - 1$ bit of Knight. Cycle the above steps, $< x_{a_1-1}, \cdots, x_0 >_3$ can be restored.

Theorem 3.3. *Suppose that γ is a generator of XTR subgroups, the order $q|p^2 - p + 1$ (q is a large prime), then for any $0 < x < p - 1$, $y \equiv Tr(\gamma^x)(\bmod p)$, all x bits can be predicted by the least significant bit.*

Corollary 3.1. *The parameter selection is the same as theorem 3, and if $q = 2^k - 1$, then each bit can be used to predict other bits of x.*

4. **Encryption and decryption process.** In this section, we send email through our encryption email system, the developing environment of email encryption systemis Visual Studio 2012, we simulate actual process under virtual machine environment. This experiment requires two computers that can be networked.

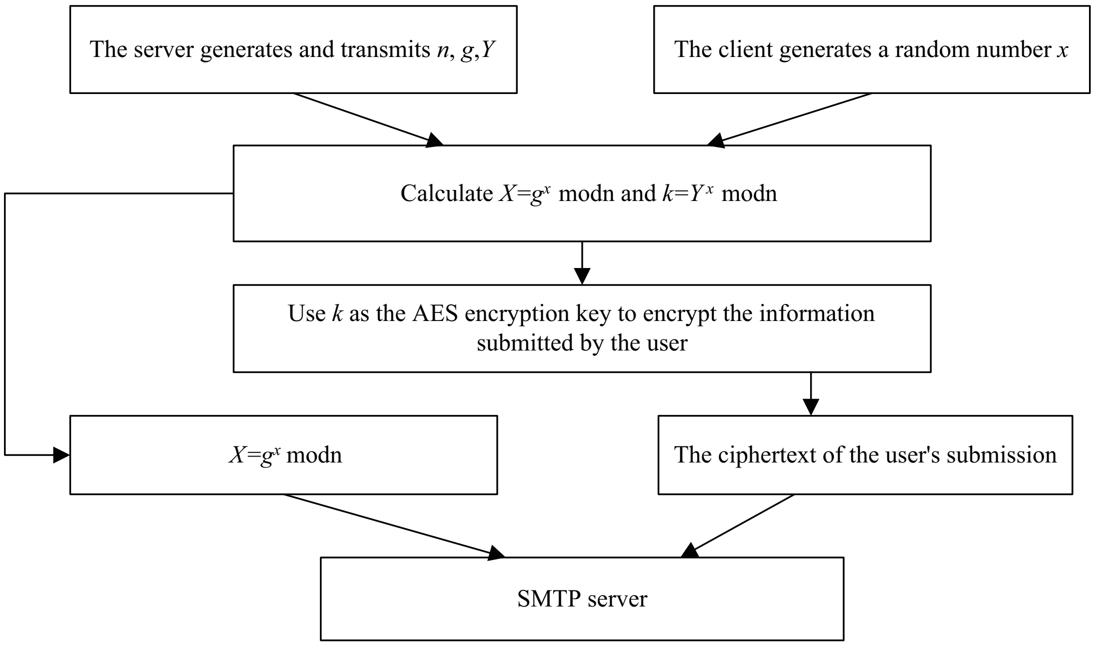

FIGURE 4. The principle of email encryption system

Encrypt message

(1) Start the encryption software on one computer and click into fcl account, enter sender's mailbox and password to create a message, where you can choose 163 and Ali cloud mailbox, enter recipient address 1174425615@qq.com.
(2) Then enter subject and content of the message, click on the encryption button when we need to encrypt theme and content. If you want to encrypt a image file, confidential documents or compressed package file, you need to click Add Attachment button to select file path to add.
(3) There are many filling modes such as PKCS7, ANSIX923, ISO10126, Zeros, we choose PKCS7 filling mode here. Encryption mode such as CBC, CFB, ECB, OFB, we choose CBC encryption mode.
(4) Finally we click Send, in this way can we sent a ciphertext message to 1174425615@qq.com, the process is shown in Figure 5.

Decrypt message

(1) We start decryption software on another machine and enter mailbox 1174425615@qq.com, we will find the ciphertext e-mail in received mail list, download file can not display completely.
(2) Click the button to select the file and add file path that need to be decrypted after download, then copy ciphertext contents to display window, fill mode and encryption mode are same as encryption.
(3) Decryption secret key is generated by DH algorithm which is same as encryption key, at last, click on decrypt content and decrypt attachment, in this way can we decrypt ciphertext message from the sender, process is shown in Figure 6.

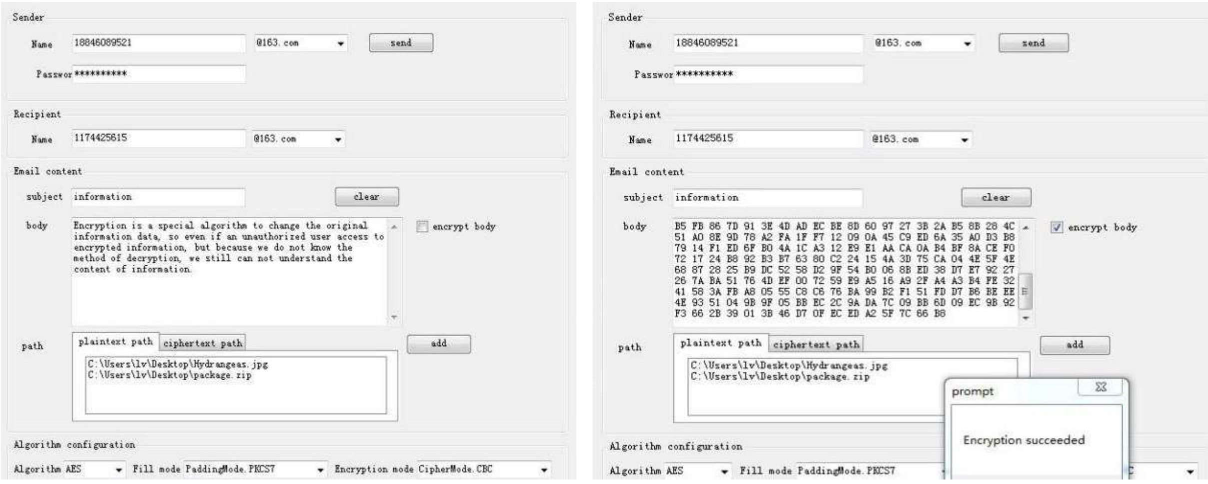

FIGURE 5. Email encryption

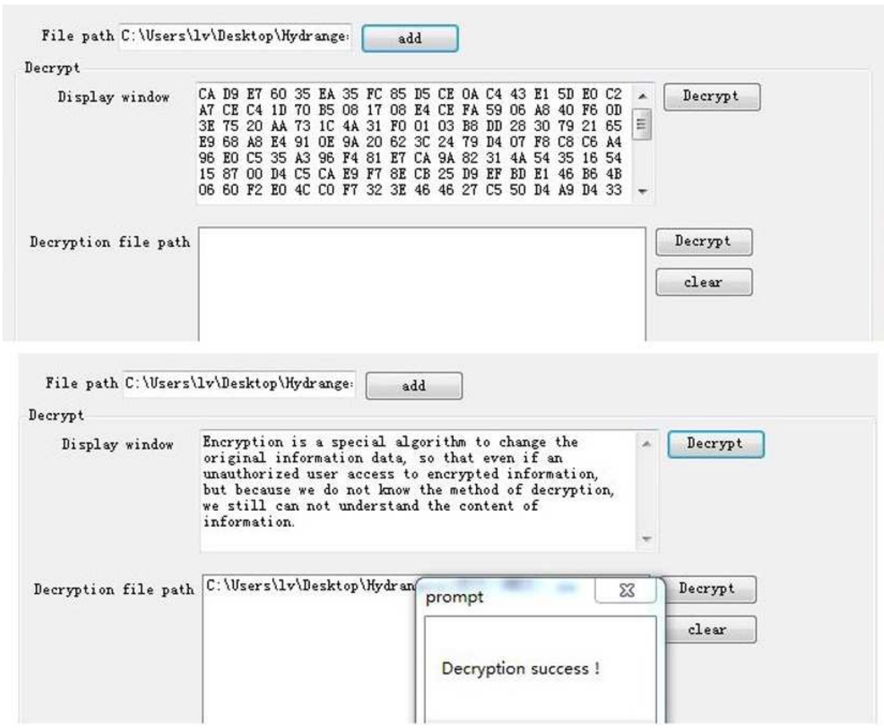

FIGURE 6. Email decryption

5. Experimental results and comparative analysis.

5.1. **AES Diffusion Confusion Test.** Diffusion and confusion are two basic ways that Shannon has proposed to design a cryptosystem to counter the adversary's statistical analysis. The diffusion is that each of plaintext influence many bits in ciphertext, so it can conceal statistical properties of plaintext; confusion is that statistical relationship between ciphertext and key as complex as possible, which makes the opponent can not release the key even if attacker get close to some statistical properties of ciphertext. In this paper, during encryption and decryption, we use same matrix, which will affect original algorithm of diffusion and confusion characteristics, this article illustrate characteristics through experiments.

Firstly, we will test the spread of 128 bit string AES encryption to ensure that key is unchanged and record the number of ciphertext bits when plaintext changes one bit, due to space reasons, here are only three changes in plaintext test results. When plaintext changes, changes caused by ciphertext of original algorithm and improved algorithm are shown in Table 1.

Table 1. The number of ciphertext bits changes when key is unchanged

Plaintext changes	Original algorithm	Improved algorithm
1 bit	65 ± 5 bits	64 ± 7 bits
2 bits	63 ± 7 bits	63 ± 5 bits
3 bits	63 ± 7 bits	64 ± 5 bits

Then, we test its confusion and record the impact of the ciphertext when key changes one bit to ensure that plaintext is unchanged. When the key changes, the changes caused by cipher text of original algorithm and improved algorithm are shown in Table 2.

Table 2. The number of ciphertext bits changes when plaintext is unchanged

Key changes	Original algorithm	Improved algorithm
1 bit	63 ± 7 bits	63 ± 5 bits
2 bits	64 ± 7 bits	64 ± 7 bits
3 bits	64 ± 5 bits	63 ± 7 bits

In this paper, a total of 30 bits plaintext and key changes were tested, and the number of ciphertext changes was about 64 bits, which indicated that improved algorithm had no effect on diffusion aliasing characteristics of the original algorithm.

5.2. **Email encryption and decryption rate test.** In this paper, we encrypt and decrypt 10000 times on five groups of 128 bit string in order to make sure experimental results more obvious, and record the time-consuming situation.

Table 3. Time-consuming of Encryption and Decryption

time/ms	Original algorithm		Improved algorithm	
	encrypt	decrypt	encrypt	decrypt
The first	470	491	348	351
The second	450	501	340	341
The third	461	490	351	353
The fourth	471	500	358	361
The fifth	460	481	349	351
Average	462.7	486.3	355.2	358.4

As can be seen from the table 3, improved AES algorithm, the acceleration of the original algorithm can be increased by 22%, decryption rate can be increased by 26% compared to the original algorithm, encryption and decryption time-consuming is equal. It shows that improved algorithm has some advantages over the original one, and it solves the problem of delay in decryption compared with encryption in original algorithm.

6. **Conclusion.** In this paper, the security of e-mail itself is based on security of encryption. The definition of security E-Mail system is that mail content is not exposed to third parties, to ensure complete and reliable E-Mail to reach the receiver, sender and receiver can know proper time to receive mail, a complete, detailed and reliable receiving proof. Therefore, DH key exchange algorithm is used to generate a symmetric key encryption required, AES algorithm is used where information is encrypted in blocks and the symmetric secret key is used for encryption and decryption process., which can enhances the anti-attack capability greatly in encryption system and guarantees the security of information transmission effectively. The safety of E-Mail system can realize the confidentiality, authentication and data integrity security function.

Acknowledgment. Project supported by the National Natural Science Foundation of China (Grant Nos. 61471158), and Project supported by the Modern sensor technology of Universities in Heilongjiang(Grant No. 2012TD007).

REFERENCES

[1] Q. Q. Chen, Y. Q. Zhang, and X. N. Li, Study and implementation of secure e-mail computer engineering, *Information Security and Communications Privacy*, vol.28, no.6, pp.121–122, 2002.
[2] Q. Li, J. P. Wu, and M. W. Xu, Forward-secure e-mail association, *Chinese Journal of Electronics*, vol. 37, no.10, pp.2302–2308, 2009.
[3] W. L. Wu, and D. G. Feng, Research status of block cipher mode, *Chinese Journal of Computers*, vol.29, no.1, pp.22–25, 2006.
[4] E. G. Zhu, G. L. Wu, and G. X. Yang. Network security protection solutions of electric power enterprise based on VPN technology, *International Conference on Computational Intelligence and Security*, pp.402–405, 2009.
[5] L. Hu, W. Yuan, and M. T. Yu, Unidirectional strategy and AES key generation algorithm improvement, *Journal of Jilin University: Engineering Science*, vol.39, no.1, pp.138–141, 2009.
[6] M. M. Kermani, and A. R. Masoleh, Fault detection structures of the S-boxes and inverse S-boxes for the advanced encryption standard, *Electronic Testing*, vol.25, no.4, pp.225–245, 2009.
[7] Y. G. Yang, and H. Z. Yu, Design of encryption chip based on RSA and AES hybrid algorithm, *Computer Knowledge and Technology*, vol.3, pp.84–86, 2006.
[8] N. N. Gao, and L. H. Song, Reconfigurable design of key expansion and column mixing unit in AES algorithm meter, *Journal of Beijing Information Science and Technology University*, vol.27, no.4, pp. 51–55, 2012.
[9] T.Y. Wu, T.T. Tsai, Y.M. Tseng, Efficient searchable ID-based encryption with a designated server, *Annals of telecommunications*, Vol. 69(7-8), pp. 391-402, 2014.
[10] T.Y. Wu, F. Meng, C.M. Chen, S. Liu, J.S. Pan, On the security of a certificateless searchable public key encryption scheme, *Proc. 10th International Conference on Genetic and Evolutionary Computing (ICGEC 2016), Genetic and Evolutionary Computing 536*, pp. 113-119, Springer.
[11] Y.M. Tseng, T.Y. Wu, T.T. Tsai, A convinced commitment scheme for bilinear Diffie-Hellman problem, *Proc. 7th International Conference on Networked Computing and Advanced Information Management (NCM 2011)*, pp. 156-161, IEEE.
[12] Y.M. Tseng, T.Y. Wu, A novel convinced Diffie-Hellman computation scheme and its cryptographic application, *Proc. The 2010 International Conference on Computational Science and Its Applications (ICCSA 2010), Lecture Note in Computer Science 6019*, pp. 225-235, 2010, Springer.
[13] G. H. Cheng, X. M. Qi, and Y. L. Luo, Polynomial modulo operation in AES algorithm and its performance analysis, *Computer Technology and Development*, vol.20, no.9, pp.115–118, 2010.
[14] X. L. Fang, W. B. Merolla, J. M. Dudley, J. M. Larger, L. Guyeux, and C. Bahi, AES contributions in fast random bit sequence generated from broadband optoelectronic entropy sources, *IEEE Transactions on Circuits and Systems*, vol.61, no.3, pp.888–901, 2014.
[15] H. Kong, and Z. H. Zheng, Analysis to several typical authenticated diffie-hellman key agreement protocols, *Computer Engineering and Applications*, vol.37, no.18, pp.72–74, 2001.