The Probabilistic Encryption Algorithm Using Linear Transformation

K. Adi Narayana Reddy[1] and B. Vishnuvardhan[2]

[1] Faculty of CSE, ACE Engineering College, Hyderabad, India
[2] Faculty of IT, JNTUH College of Engineering, Karimnagar, India

Abstract. The probabilistic encryption produces more than one ciphertext for the same plaintext. In this paper an attempt has been made to propose a probabilistic encryption algorithm based on simple linear transformation. The variable length sub key groups are generated using a random sequence. A randomly selected element is replaced by each element of the plaintext from the corresponding indexed sub key group. With this a cryptanalyst cannot encrypt a random plaintext looking for correct ciphertext. The security analysis and performance of the method are studied and presented.

Keywords: Hill cipher, Sub key groups, Pseudo Random number, Probabilistic encryption.

1 Introduction

The information to be transmitted must be secure against several attacks. This is achieved by using encryption and decryption techniques, which converts readable of information into unreadable form and vice versa. Many numbers of cryptographic algorithms are available to provide secured transformation of information, but the efficiency and strength of the algorithm is one of the most important aspects to be studied in the field of information security. With the development of probabilistic encryption algorithms a cryptanalyst cannot encrypt random plaintext looking for correct ciphertext because the encryption process produces more than one ciphertext for one plaintext. We consider linear transformation based cryptosystem which is a simple classical substitution cipher.

In 1929 Hill developed a simple cryptosystem based on linear transformation. It is implemented using simple matrix multiplication and it hides single character frequency and also hides more frequency information by the using large key matrix. But it is vulnerable to known plaintext attack and the inverse of every shared key matrix may not exist all the time. It is a simple traditional symmetric key cipher and the message is transmitted through the communication channel is divided into 'm' blocks, each of size 'n'. Assume that both 'n' and 'm' are positive integers and M_i is the i^{th} block of plaintext. This procedure encrypt each of the block M_i, one at a time using secret key matrix. It maps each character with unique numeric value like A=0, B=1 ... to produce the 'n' characters in each of the block. The i^{th} ciphertext block C_i can be obtained by encrypting the i^{th} plaintext block M_i using the following equation (1)

S.C. Satapathy et al. (eds.), *Emerging ICT for Bridging the Future – Volume 2,*
Advances in Intelligent Systems and Computing 338, DOI: 10.1007/978-3-319-13731-5_42

$$C_i = M_i K \bmod m \tag{1}$$

In which K is an n x n key matrix. The plaintext can be obtained from the decrypted cipher text using following equation (2)

$$M_i = C_i K^{-1} \bmod m \tag{2}$$

In which K^{-1} is the key inverse and it exist only if the GCD (det K (mod m), m) = 1.

Many researchers improved the security of linear transformation based cryptosystem. Yeh, Wu et al. [16] presented an algorithm which thwarts the known-plaintext attack, but it is not efficient for dealing bulk data, because too many mathematical calculations. Saeednia [13] presented an improvement to the original Hill cipher, which prevents the known-plaintext attack on encrypted data but it is vulnerable to known-plaintext attack on permutated vector because the permutated vector is encrypted with the original key matrix. Ismail [5] tried a new scheme HillMRIV (Hill Multiplying Rows by Initial Vector) using IV (Initial Vector) but Rangel-Romeror et al. [10] proved that If IV is not chosen carefully, some of the new keys to be generated by the algorithm, may not be invertible over Z_m, this make encryption/decryption process useless and also vulnerable to known-plaintext attack and also proved that it is vulnerable to known-plaintext attack. Lin C.H. et al. [8] improved the security of Hill cipher by using several random numbers. It thwarts the known-plaintext attack but Toorani et al.[14, 15] proved that it is vulnerable to chosen ciphertext attack and he improved the security, which encrypts each block of plaintext using random number and are generated recursively using one-way hash function but Liam Keliher et al [7] proved that it is still vulnerable to chosen plaintext attack . Ahmed Y Mahmoud et al [1, 2, 3] improved the algorithm by using eigen values but it is not efficient because the time complexity is more and too many seeds are exchanged. Reddy, K.A. et al [11, 12] improved the security of the cryptosystem by using circulant matrices but the time complexity is more. Again Kaipa, A.N.R et al [6] improved the security of the algorithm by adding nonlinearity using byte substitution over GF (2^8) and simple substitution using variable length sub key groups. It is efficient but the cryptanalyst can find the length of sub key groups by collecting pair of same ciphertext and plaintext blocks. Later Adinarayana reddy [17] improved the security of [6] by dynamic byte substitution where the dynamic byte substitution shifts the static location to new secret location. Goldwasser and Micali introduced the probabilistic encryption algorithm [18] but is impractical to implement. In this paper randomness will be included to the linear transformation based cryptosystem to overcome chosen-plaintext and chosen-ciphertext attacks and to reduce the time complexity.

In this paper an attempt has been made to introduce the concept of probabilistic encryption. The detailed algorithm is presented in section-2 and its performance and security analysis are studied and presented in section-3.

2 Method

In this paper an attempt is made to propose a randomized encryption algorithm which produces more than one ciphertext for the same plaintext. The following sub sections explain the proposed method.

2.1 Algorithm

Let M be the message to be transmitted. The message is divided into 'm' blocks each of size 'n' where 'm' and 'n' are positive integers and pad the last block if necessary. Let M_i be the i^{th} partitioned block (i = 1, 2, ... m) and size of each M_i is 'n'. Let C_i be ciphertext of the i^{th} block corresponding to the i^{th} of block plaintext. In this paper the randomness is added to the linear transformation based cryptosystem. Each element of the plaintext block is replaced by a randomly selected element from the corresponding indexed sub key group. The randomly selected element will not be exchanged with the receiver. In this method key generation and sub key group generation is similar to hybrid cryptosystem [3]. Choose a prime number 'p'. The following steps illustrate the algorithm.

1. **Step 1: Key Generation.** Select randomly 'n' numbers $(k_1, 1k_2 \cdots, k_n)$ such that $\mathrm{GCD}(k_1, 1k_2 \cdots, k_n) = 1$. Assume $k_i \in Z_p$. Rotate each row vector relatively right to the preceding row vector to generate a shared key matrix K_{nxn}. The generated key matrix is called prime circulant matrix.
2. **Step 2: Sub Key Group Generation.** Let r = $\sum_{i=1}^{n} k_i$ mod p and generate a sequence of 'p' pseudo-random numbers S_i (i = 0, 1,… p-1) with initial seed value as r. Generate the sub-key group S_G as i from pseudo-random numbers as, j = (i + S_i) mod b, for all $i \in S_{G_J}$ and $b < \lfloor p/2 \rfloor$. (i.e. the sub-key groups are formed with the pseudorandom number sequence.)
3. **Step 3: Encryption.** The encryption process encrypts each block of plaintext using the following steps.
   - 3.1 Initially the transformation is applied as $Y = KM \bmod p$
   - 3.2 The index of very element of the vector Y is calculated as, Index = Y mod b (i.e. Y mod b = (y_1 mod b, … y_n mod b)) and corresponding to the vector Index an element is randomly selected from the corresponding sub key group S_G and that becomes C_2 and C_1 = (y_1 / b, … y_n / b) + (y_1 mod b, … y_n mod b) mod p i.e. C1 = Y/b + Index
   - 3.3 The pair of ciphertext (C_1, C_2) is transferred to other end.
4. **Step 4: Decryption.** The decryption process decrypts each of the received ciphertext pair (C1, C2) using the following steps

4.1 Receiver receives ciphertext pair (C_1, C_2) and searches for the index vector Index from C_2 then calculate Y as Y = (b*(C_1-Index) + Index) mod p

4.2 In order to obtain the plaintext the inverse key matrix is multiplied with the resultant vector Y as

$$M = K^{-1}Y \bmod p$$

The algorithm is explained through the following example

2.2 Example

Consider a prime number p as 53 and the set of relatively prime numbers as [5, 27, 13]. Generate shared key matrix K_{3x3}. Assume the plaintext block M= [12, 14, 3]. Generate a sequence of 'p' pseudo-random number with seed value as r = 45. Assume b = 5 and generate five sub-key groups (S_G) from the random number sequence. The sub key groups are random and of variable length.

S_G [0] = {0, 6, 17, 21, 24, 25, 31, 38, 50}
S_G [1] = {1, 4, 9, 12, 16, 29, 30, 34, 39, 40, 43, 44, 46, 48, 49}
S_G [2] = {2, 3, 13, 22, 23, 26, 37, 45, 51, 52}
S_G [3] = {7, 10, 15, 19, 20, 27, 33, 42}
S_G [4] = {5, 8, 11, 14, 18, 28, 32, 35, 36, 41, 47}
Y = KM mod p = KM mod 53 = [0, 42, 44]
Y/ b = (0, 8, 8)
Y% b = (0, 2, 4)

Now select elements randomly from the corresponding sub key groups and add the position of those elements to the corresponding quotient Y / b.

The possible ciphertext pairs are presented in table 1.

The same plaintext is mapped to many ciphertext pairs

After communicating the ciphertext pair (C_1, C_2) to the receiver, the decryption process outputs the plaintext as [12, 14, 3].

3 Performance Analysis

The performance analysis is carried out by considering the computational cost and security analysis which are to show the efficiency of the algorithm.

3.1 Computational Cost

The time complexity measures the running time of the algorithm. The time complexity of the proposed algorithm to encrypt and to decrypt the text is O (mn^2) which is shown in the equation (4), where 'm' is number of blocks and 'n' is size of each block, which is same as that of original Hill cipher. In this process T_{Enc} and T_{Dec} denote the running time for encryption and decryption of 'm' block of plaintext respectively.

$$T_{Enc}(m) \cong m(n^2)T_{Mul} + m(n^2)T_{Add}$$
$$T_{Dec}(m) \cong m(n^2)T_{Mul} + m(n^2)T_{Add} + mnT_s \quad (3)$$

In which T_{Add}, T_{Mul}, and T_s are the time complexities for scalar modular addition, multiplication, and search for the index respectively.

$$T_{Enc}(m) \cong m(n^2)c_1 + m(n^2)c_2 \cong O(mn^2)$$
$$T_{Dec}(m) \cong m(n^2)c_1 + m(n^2)c_2 + mnc_3 \cong O(mn^2) \quad (4)$$

Where c_1,c_2 and c_3 are the time constants for addition, multiplication and index search respectively. The running time of encryption our method and other methods are analysed and presented in the Fig 1. The running time our method is equal to the linear transformation based cipher. As the block size increases the time to encrypt is linearly increasing whereas other method is increasing exponentially. Similarly the time to decrypt also increases linearly as block size increases. This method outperforms comparing with other methods. Our method even reduced the space complexity because it needs only n memory locations to store the key as the shared key is circulant matrix.

3.2 Security Analysis

The key matrix is shared secretly by the participants. The attacker tries to obtain the key by various attacks but it is difficult because the random selection of elements from sub key groups. It is difficult to know the elements of the sub key groups because each sub key group is of variable length and generated by modulo which is an one-way function.

The proposed cryptosystem overcomes all the drawbacks of linear transformation based cipher and symmetric key algorithms. This is secure against known-plaintext, chosen-plaintext and chosen-ciphertext attacks because one plaintext block is encrypted many number of ciphertext blocks. This is due to the random selection of element from the corresponding sub key group. Therefore, the cryptanalyst can no longer encrypt a random plaintext looking for correct ciphertext. To illustrate this assume that the cryptanalyst has collected a ciphertext C_i and guessed the corresponding plaintext M_i correctly but when he/she encrypt the plaintext block M_i the corresponding ciphertext block C_j will be completely different. Now he/she cannot confirm M_i is correct plaintext for the ciphertext C_i. This makes the more secure and efficient. It also requires huge memory. It is also secure against brute force attack if the key size is at least 8 and prime modulo p is at least 53. It is free from all the security attacks.

Table 1. Ciphertext corresponding to Plaintext

	Plaintext M	C1	C2
Case 1	[12, 14, 3]	[0, 10, 12]	[6, 26, 41]
Case 2	[12, 14, 3]	[0, 10, 12]	[31, 51, 18]
Case 3	[12, 14, 3]	[0, 10, 12]	[50, 2, 18]

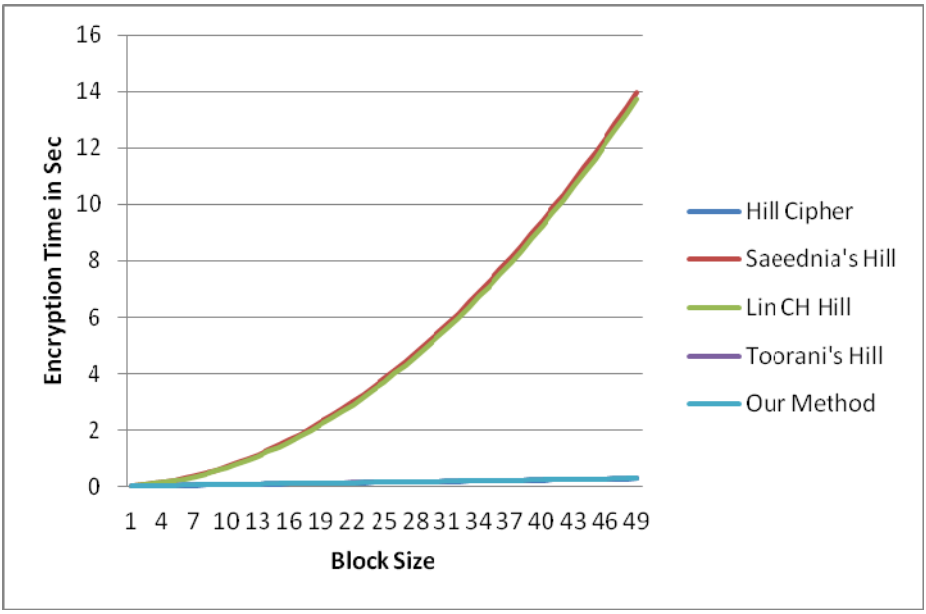

Fig. 1. Block Size Vs Encryption time

4 Conclusion

The structure of the proposed cryptosystem is similar to substitution ciphers i.e. initially the linear transformation is applied on the original plaintext block then the result is replaced by a randomly selected element from the corresponding sub key group. The sub key groups are of variable length and each sub key group is generated randomly using one-way modulo function. The proposed probabilistic encryption algorithm produces more than one ciphertext for one plaintext because each element of the block is replaced by a randomly selected element from the corresponding sub key group. The proposed cryptosystem is free from all the security attacks and it has reduced the memory size from n^2 to n, because key matrix is generated from the first row of the matrix and is simple to implement and produces high throughput.

References

1. Ahmed, Y.M., Chefranov, A.G.: Hill cipher modification based on eigenvalues hcm-EE. In: Proceedings of the 2nd International Conference on Security of Information and Networks, October 6-10, pp. 164–167. ACM Press, New York (2009), doi:10.1145/1626195.1626237
2. Ahmed, Y.M., Chefranov, A.: Hill cipher modification based on pseudo-random eigen values HCM-PRE. Applied Mathematics and Information Sciences (SCI-E) 8(2), 505–516 (2011)
3. Ahmed, Y.M., Chefranov, A.: Hill cipher modification based generalized permutation matrix SHC-GPM. Information Science Letter 1, 91–102
4. Hill, L.S.: Cryptography in an Algebraic Alphabet. Am. Math. Monthly 36, 306–312 (1929), http://www.jstor.org/discover/10.2307/2298294?uid=3738832&uid=2129&uid=2&uid=70&uid=4&sid=21102878411191
5. Ismail, I.A., Amin, M., Diab, H.: How to repair the hill cipher. J. Zhej. Univ. Sci. A. 7, 2022–2030 (2006), doi:10.1631/jzus.2006.A2022
6. Kaipa, A.N.R., Bulusu, V.V., Koduru, R.R., Kavati, D.P.: A Hybrid Cryptosystem using Variable Length Sub Key Groups and Byte Substitution. J. Comput. Sci. 10, 251–254 (2014)
7. Keliher, L., Delaney, A.Z.: Cryptanalysis of the toorani-falahati hill ciphers. Mount Allison University (2013), http://eprint.iacr.org/2013/592.pdf
8. Lin, C.H., Lee, C.Y., Lee, C.Y.: Comments on Saeednia's improved scheme for the hill cipher. J. Chin. Instit. Eng. 27, 743–746 (2004), doi:10.1080/02533839.2004.9670922
9. Overbey, J., Traves, W., Wojdylo, J.: On the keyspace of the hill cipher. Cryptologia 29, 59–72 (2005), doi:10.1080/0161-110591893771
10. Rangel-Romeror, Y., Vega-Garcia, R., Menchaca-Mendez, A., Acoltzi-Cervantes, D., Martinez-Ramos, L., et al.: Comments on "How to repair the Hill cipher". J. Zhej. Univ. Sci. A 9, 211–214 (2008), doi:10.1631/jzus.A072143
11. Reddy, K.A., Vishnuvardhan, B., Madhuviswanath, Krishna, A.V.N.: A modified hill cipher based on circulant matrices. In: Proceedings of the 2nd International Conference on Computer, Communication, Control and Information Technology, February 25-26, pp. 114–118. Elsevier Ltd. (2012), doi:10.1016/j.protcy.2012.05.016
12. Reddy, K.A., Vishnuvardhan, B., Durgaprasad: Generalized Affine Transformation Based on Circulant Matrices. International Journal of Distributed and Parallel Systems 3(5), 159–166 (2012)
13. Saeednia, S.: How to make the hill cipher secure. Cryptologia 24, 353–360 (2000), doi:10.1080/01611190008984253
14. Toorani, M., Falahati, A.: A secure variant of the hill cipher. In: Proceedings of the IEEE Symposium on Computers and Communications, July 5-8, pp. 313–316. IEEE Xplore Press, Sousse (2009), doi:10.1109/ISCC.2009.5202241
15. Toorani, M., Falahati, A.: A secure cryptosystem based on affine transformation. Sec. Commun. Netw. 4, 207–215 (2011), doi:10.1002/sec.137
16. Yeh, Y.S., Wu, T.C., Chang, C.C.: A new cryptosystem using matrix transformation. In: Proceedings of the 25th IEEE International Carnahan Conference on Security Technology, October 1-3, pp. 131–138. IEEE Xplore Press, Taipei (1991), doi:10.1109/CCST.1991.202204
17. Reddy, K.A., Vishnuvardhan, B.: Secure Linear Transformation Based Cryptosystem using Dynamic Byte Substitution. International Journal of Security 8(3), 24–32
18. Gold Wasser, S., Micali, S.: Probabilistic Encryption. Journal of Computer and System Sciences 28(2), 270–299, doi:10.1016/0022-0000(84)90070-9