ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 8(14), 59192, July 2015* ISSN (Online) : 0974-5645

Security Analysis and Modification of Classical Encryption Scheme

Maya Mohan[1*], M. K. Kavitha Devi[2] and V. Jeevan Prakash[3]

[1]Department of CS&E, NSSCE, Palakkad - 678008, Kerela, India; mayajeevan@gmail.com
[2]Department of CSE, TCE, Madurai - 625015, Tamilnadu, India; mkkdit@tce.edu
[3]Department of Mathematics, NSSCE, Palakkad - 678008, Kerela, India; jeevanprakashv@gmail.com

Abstract

Objectives: Computer security is all about the study of cyber attacks with a view to defend against them. Cryptography is considered to be a class of science by using the special art of transforming information in a protected way such that it can overcome the attacks. There is an immense requirement of strong cryptographic algorithms in order to withstand against the various attacks. **Methods**: According to the Kerchoff''s Principle one should always assume that the adversary knows the encryption and decryption algorithms. The resistance of the cipher to attack must be based only on the secrecy of the key. There comes the play of cryptanalysis. It is art of breaking the keys by identifying the vulnerabilities existing in the systems. This paper deals with the classical encryption schemes and their cryptanalysis. The cryptanalysis for various encryption schemes differs alot. Various cryptanalysis like statistical analysis, frequency analysis, trial and error (brute force) are carried out in this work. **Findings**: The classical encryption schemes such as caesar cipher, shift cipher, vigenere cipher, affine cipher and hill cipher are discussed in the paper. A detailed analysis about the security of the above mentioned ciphers are explored. Among the ciphers it is identified that if the key varies for each plaintext to be encrypted provides added security. But the worst is the management of the huge key space. A modified algorithm is proposed which can provide a better security using simple computations. In this approach lots of keys are used but generated from a single key by using simple shift and EXOR operation. In the main stream only one key need to be exchanged between the communications entities and for that exchange we can make use of the public key cryptosystem. **Application**: Cryptography is considered to be an ineluctable field in era of communication. Cryptographic algorithms act as an underpinning for lots of applications such as Anonymous Remailers, Digital Signatures, Secured Money transactions etc.

Keywords: Additive Cipher, Monoalphabetic Cipher, Multiplicative Cipher, Polyalphabetic Cipher

1. Introduction

Cryptography[1] plays a vital role in the epoch of communication especially in e-transactions. A wide variety of innovations are emerged in the area of cryptography to achieve different levels of security. Cryptography is mainly classified based on the Number of keys used (Hash functions: no key, Secret key cryptography: one key, Public key cryptography: two keys - public, private), classification of encryption operations used (substitution / transposition / product) and the method for processing the plaintext (block/stream).The cryptanalysis[2] basically work on the above parameters. Based on the information needed by the cryptanalyst attacks can be classified into four broad categories.

- Ciphertext only attack- The only information known to the cryptanalyst is the ciphertext
- Known plaintext attack- given ciphertext , prior knowledge about ciphertext -plaintext pairs
- Chosen plaintext attack- a given ciphertext ,by using encryption algorithm with plaintexts and getting the matching ciphertexts
- Chosen ciphertext attack (most severe)- a given ciphertext, by using decryption algorithm with ciphertexts and getting the matching plaintexts. (reverse of the previous one)

* *Author for correspondence*

In cryptanalysis, frequency analysis is the study of the frequency of letters or groups of letters in a ciphertext. The method is used as an aid to breaking classical ciphers. There are other factors which influence the cryptanalysis. It includes the computational resources required to break the cipher (like the time needed, memory needed etc) and the amount and the quality of the data recovered from cryptanalysis. Different categories of attacks are given below.

1. Ciphertext-Only Attack (COA) - In this type of attack the cryptanalyst is having only the ciphertext. This is normally happened in real life cryptanalysis, but it is considered to be the weakest attack because of the cryptanalyst's lack of information. Modern ciphers need to be very resistant to this category of attack.
2. Brute Force Attack or Exhaustive Key Search (BFA) - In this type of attack all the possible key is tried until the retrieval of the original key. All ciphers existing today except the one time pad is vulnerable brute force attack. Security depends both on the cipher as well as the length of the key. If the key is N bits in length, then there will be are 2^N possible keys to check , so the BFA can break the cipher in the worst-case equal to 2^N and an average case equal to 2^{N-1} This is considered as a standard of comparison for other attacks.
3. Known-Plaintext Attack (KPA) - This attack is assumed that the pairs of plaintext and the corresponding cipher text are available to the cryptanalyst, the key used to encrypt the plaintext is publicly available, allowing the cryptanalyst to create the ciphertext for any plaintext. So all the public-key encryption algorithms must be very resistant to all the known-plaintext attacks.
4. Chosen-Plaintext Attack (CPA) - In this attack the cryptanalyst is allowed to select a number of plaintexts to be encrypted and have the right access the ciphertext. This allows to explore over the plaintext to exploit vulnerabilities and non random behaviour which found only with particular plaintexts.
5. Chosen-Ciphertext Attack (CCA) - In this attack the cryptanalyst can choose randomly any ciphertext and have access to the corresponding plaintext. In real life this would require the analyst to have access to the medium and the receiver.
6. The Wired Equivalent Privacy (WEP) - Privacy protocol is used to protect Wi-Fi internet devices is vulnerable to key search attacks.
7. Side Channel Attack - This is neither a type of cryptanalytic attack, nor depend on the strength of the key or the algorithm. It uses other data about the encryption or decryption process to gain information about the message, such as electronic noise produced by encryption machines, or the sound produced by keystrokes when the plaintext is typed, or by measuring the computations taken place.

Different types of attacks are existing in other cryptographic primitives, or other security systems[10]. Example for such type of attack is Adaptive chosen-message attack for digital signatures[9].

2. Cryptanalysis of Classical Encryption Schemes

According to Kerchoff's Principle one should always assume that the adversary knows the encryption and decryption algorithms. The resistance of the cipher to attack must be based only on the secrecy of the key.

2.1 Monoalphabetic Ciphers

In Monoalphabetic substitution a character in the plaintext is always replaced by the same character in the ciphertext regardless of its position in the text. The relationship between a character in the plaintext to a character in the ciphertext is always one to one. The cryptanalysis can be done as follows.

Step 1: The text encrypted using substitution is given as input.
Step 2: Calculate the frequency of every letter of the ciphertext.
Step 3: Replace the letter having highest frequency with highest frequency letter standard of English.
Step 4: Repeat the process for the remaining letters in the descending order of the frequency.
Step 5: Compare the result with standard dictionary.
Step 6: If a match found, output the plain text else shift the index of the calculated frequency and go to step 3.

Monoalphabetic cipher mainly are of two types: Additive cipher and Multiplicative cipher

2.1.1 Additive Cipher

In Additive cipher the plaintext, cipher text and the key are integers in Z_{26}. It is classified in two. They are Shift cipher and Caesar cipher. In Shift cipher the user can select any key where as in Caesar cipher the key is fixed and its value is 3. The encryption and decryption for both ciphers are given in Table 1.

Table 1. Encryption and Decryption Process

Algorithm	Encryption	Decryption
Shift cipher	C=P + K mod 26	P=C-K mod 26
	K= 0...25	K=0...25
Caeser	C=P+K mod 26	P=C-K mod 26
Cipher	K=3	k=3

2.1.2 Cryptanalysis of Additive Cipher

The additive ciphers are vulnerable to cipher text only attacks using exhaustive key search methods. The main attacks are brute force attack (trying all possible keys i.e. 0-25) and statistical attack (observing for similar pattern for recovering the plaintext i.e. AA, BBB etc). The plaintext will be computed with an average case after trying 26/2 = 13 times. The cryptanalysis for caesar cipher with unknown key is given in Table 2.

Table 2. Caeser Cipher Cryptanalysis

Key	Number of Alternative Keys	Time required at 1 decryption/μs	Time required at 106 decryptions/μs
26characters (permutation)	26! = 4 *10^{26}	2 * 10^{26} μs = 6.4* 10^{12} years	6.4* 10^{6} years

Cryptanalysis can be performed by frequency analysis of the English characters because it is more vulnerable to frequency analysis attacks. Each language has certain features: frequency of letters, or of groups of two or more letters. Normally Substitution ciphers preserve the language features. This property of the language is make use for cryptanalysis. The number of different ciphertext characters or combinations are counted to determine the frequency of usage. The cipher text is examined for patterns, repeated series, and common combinations. Replacement of ciphertext characters with possible plaintext equivalents using known language characteristics. The time needed to perform brute force attack is proportional to key space. The time required to perform statistical (frequency) analysis for ciphers with different size is given in the Figure 1.

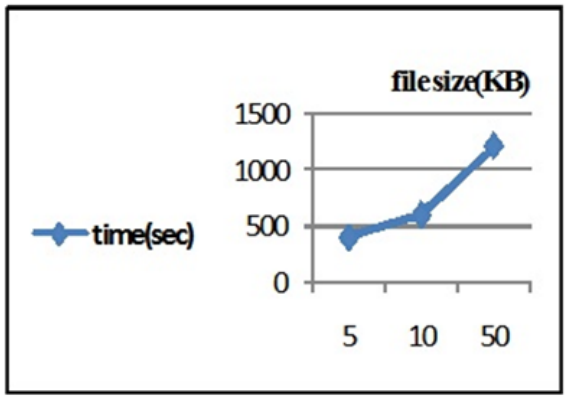

Figure 1. Statistical cryptnalysis.

2.1.3 Security Improvements to Additive Cipher

Using null values- By adding null values in between to confuse the cryptanalyst. Misspells words- deliberately misspelling the words in the plain text. In order to escape from frequency analysis, we make use of homophonic substitution cipher

A one - many mapping of symbols is performed. Substitution is said to add confusion

e.g. 0 → {01, 10}, 1→{00,11}

Advantage: character frequencies are hiding

Disadvantage: The length of the message and key is very long

2.2 Multiplicative Cipher

To convert a plain letter P to the cipher letter C using the Multiplication Cipher, we use the encryption function: f: P → C = (a * P) MOD 26. If *a* is a wisely chosen key, that is if *a* is relatively prime to 26, then the function f produces a one-to-one relationship between plain and cipher letters, which therefore allows a unique encryption. For decryption the function used is f: C→ P = (C * a^{-1}) MOD 26.

Affine Cipher is a well known example for multiplicative cipher. Similar to additive cipher the main attacks on multiplicative cipher are brute force attack and frequency analysis attack.

2.2.1 Affine Cipher

An encryption scheme (or algorithm) of the form y = (ax + b) MOD 26 is called Affine cipher[5]. In the above equation x is the numerical equivalent of the given plaintext letter, and a and b are (randomly chosen) integers and there should be an inverse exists for a in 26. The decryption can be performed by the function $x=a^{-1}(y-b) \bmod 26$.

Suppose $m = \prod_{i=1}^{n} p_i^{e_i}$ then Euler's totient function

ϕ(m) is defined in eq.1

$$\varphi(m) = \prod_{i=1}^{n} (p_i^{e_i} - p_i^{e_i-1}) \quad (1)$$

The number of keys possible in affine cipher can be calculated using eq. 2

Number of Keys = m* ϕ(m) (2)

2.3 Cryptanalysis of Affine Cipher

Ciphertext only attack is possible with affine cipher. Brute force attack and statistical attack are able to perform once the cipher text is available. It is more easy to do cryptanalysis once the ciphertext-plaintext pairs are available. From the available ciphertext plaintext pairs, linear equations are formed and when solving the unknowns the keys are getting. Using the keys the remaining ciphertext can be decrypted. The key space with the Affine cipher is 312 (not 252 since some of the pairs are unusable).

2.4 Different Procedures for Breaking and Affine Cipher

Exhaustic Search-Consider there are 12 possible multiplicative parameters b such that gcd (b,26) = 1 and 26 possible additive parameters a. This gives 12 * 26 =312 total (b, a) pairs to test.

Frequency analysis- One of the easiest way which involves matching to the most frequently occurring ciphertext letters with large frequently occurring plaintext letters. It involves solving a system of equations mod 26.

2.5 Advantages and Disadvantages of Affine Cipher

Two keys are using, added security compared to additive cipher but still vulnerable to statistical attack because of the possibility of the frequency analysis of the language.

2.6 Polyalphabetic Substitution Cipher

In a polyalphabetic substitution cipher, multiple simple substitution ciphers are used to provide more security. Unlike substitution ciphers the mapping in polyalphabetic substitution ciphers is one to many, i.e. one character in the plaintext may map to many characters while doing the encryption. The particular one used changes with the position of each character of the plaintext

There are multiple one-letter keys. The first key encrypts the first letter of the plaintext, the second key encrypts the second letter of the plaintext, and so on. After all keys are used, you start over with the first key. The number of keys determines the period of the cipher. There are lot of algorithms based on polyalphabetic cipher like vigenere cipher, autokey cipher, hill cipher etc.

2.6.1 Vigenere Cipher

For a given plaintext letter P_i, the substitution value C_i is calculated, as follows: $C_i = (P_i + k_i) \bmod 26$, where k_i is the value of the letter found in the i-th position of the key sequence. Unfortunately, the Vigenère cipher is also not very secure, Code can be broken by analyzing the period. The Vigenere cipher was considered to be completely unbreakable for hundreds of years, and indeed, if very long keys are used the vigenere cipher can be totally unbreakable. But the usage of short keys, or if we have a lot of ciphertext compared to the key length, the vigenere cipher is quite easy to solve.

2.6.2 Cryptanalysis of Vigenere Cipher

Cryptanalysis of the Vigenere cipher mainly include 2 steps: First is to detect the period of the cipher (ie the length of the key), second find the key used. Given only the ciphertext, we must find the Plaintext and the key. The first step in doing this is finding the key length. Two ways to find this key length is:

- Friedman Attack implementation
- Kasiski Attack implementation

2.6.2.1 Friedman Test

The goal is to find the key length. The step as follows:

1. The character frequencies are monitored and counted to check the occurrences of each letter appears in the ciphertext.

2. Multiply each letter count by count minus 1 and then add up the sum.
3. The sum of the frequencies are computed as follows:

```
{ int i=0;
  do increment i till i < 26
{sum = sum + FC[i]*(FC[i]-1);}}
```

4. Calculating the index of coincidence as follows: Dividing the entire sum of the frequencies with the length of the cipher times the length minus1. index= sum/(lc * (lc-1)); where lc is the length of the cipher
5. The key length (KL) is calculated as KL = ((0.0265*lc)/((0.065-index) +(lc*(index-0.0385))));

2.6.2.2 Kasiski Test

1. Search the ciphertext for pairs of identical segments (length at least 3)
2. Record the distance between the starting positions of the 2 segments
3. If we obtain several such distances $d_1, d_2, \ldots$, we would conjecture that the key length m divides all of the d_i's m divides the gcd of the d_i's

Suppose $X=x_1x_2\ldots x_n$ is a string of n alphabetic characters

Index of coincidence of X, denoted $I_C(x)$: the probability that 2 random elements of X are identical.

The calculation is shown in eq. 3.We denote the frequencies of A,B,..,Z in X by $f_0, f_1, \ldots, f_{25}$.

$$I_C(X) = \frac{\sum_{i=0}^{25} \binom{f_i}{2}}{\binom{n}{2}} = \frac{\sum_{i=0}^{25} f_i(f_i - 1)}{n(n-1)} \quad (3)$$

4. Using the expected probabilities[3] P_0 - P_{25}: the expected probability of A-Z Suppose a ciphertext $Y=y_1y_2\ldots y_n$, define m substrings of $Y_1, \ldots, Y_m$ of Y

$$Y_1 = y_1 y_{m+1} y_{2m+1} \cdots$$
$$Y_2 = y_2 y_{m+2} y_{2m+2} \cdots$$
$$\vdots \; \vdots \; \vdots$$
$$Y_m = y_m y_{2m} y_{3m} \cdots$$

Each value IC (Yi) should be roughly equal to 0.065.

5. If m is not the length of the keyword and Y_i appears to be random. A completely random string will have I_c given in eq. 4

$$I_c \approx 26\left(\frac{1}{26}\right)^2 = \frac{1}{26} = 0.038 \quad (4)$$

Advantage-Very fast Less chance for propagating the error and the effects of error is very less.

Disadvantage- Each character can be analyzed separately.

2.6.3 Autokeying Cipher

The Autokey Cipher[6] is almost identical to the Vigenere Cipher. It is more secure compare to vigenere cipher. In this cipher the key consists of a collection of subkeys in which each subkey is used to encrypt the corresponding character in the plain text. The first subkey is an agreed value between the communicating parties which is secretly shared. The second subkey is the value of the first plain text character. Encryption and decryption steps are shown in eq. 5 and eq. 6 respectively

P = P1, P2, P3.... K = K1, P1, P2.....C = C1, C2, C3.....

Encryption: $C_i = P_i + K_i \bmod 26$ (5)
Decryption: $P_i = C_i - K_i \bmod 26$ (6)
Possible vulnerabilities

- Knowing the keyword can recover the first few letters
- Still have frequency characteristics to attack

2.6.4 Cryptanalysis of Auto Key Cipher

Vulnerable to brute-force attack as it comes under the category of additive cipher. The very first sub-key will be selected from one of the 25 values. Since the key is part of English Language, make use of short English words along the length of the cipher text could reveal likely English text. This will be used to guess the length of the keyword and that helps in knowing the key. Use a common small word and use trial and error for the key. Look for a meaningful English text in the resulting plaintext. Use this to guess for the length of the keyword. Shift the likely result back to find the keyword at the beginning of the shifted plaintext.

2.6.5 Hill Cipher

Hill cipher encrypts blocks of data and the block size depends on key matrix used for encryption. Uses Linear Algebra to encrypt and decrypt the data. Two key matrices

are used one for encryption and another for decryption, among them one is the inverse of other with respect to mod 26. The matrices must be of order rn x n.

The Hill cipher is a generalization of the permutation cipher (permute the letters within each block)

The encryption is as follows

C=K*P where C,P and K are matrices and K is the key matrix and it should have an inverse mod 26 to recover the plain text $P=C*K^{-14}$.

2.6.6 Cryptanalysis of Hill Cipher

- - Hard with ciphertext-only attacks
- - Easy with known plaintext cipher text attacks. Once the plain text cipher text pairs are available it is easy to find the key by solving the linear equation. On the derival of the key, with the cipher text the entire plaintext can be decrypted.

Cryptanalysis is easy with small key space, once the key matrix is of higher order the task will become more difficult. Performance evaluation of various encryption standards are included in[7].

3. A Proposed Classical Encryption Scheme

In the previous section various classical encryption algorithms are described and their cryptanalysis also done. The paper aims to propose a new algorithm in the class of monoalphabetic substitution cipher. The proposed algorithm is a modification to Vigenere Cipher[8]. It could be able to withstand the frequency analysis test and Kasiski Test. The algorithm described in the following section.

3.1 Algorithm

It is quite similar to the encryption scheme used in Vigenere Cipher. The characters in the plain text will be mapped to the integer values as given in Table 3.

Table 3. Character Mapping

Character	a	b	c	d	E	f	g	h	i	j	k	l	m
Value	0	1	2	3	4	5	6	7	8	9	10	11	12
Character	n	o	p	q	R	s	t	u	v	w	x	y	z
Value	13	14	15	16	17	18	19	20	21	22	23	24	25

In Vigenere Cipher the key will be randomly selected not based on the length of the plaintext. If the key length is less than the length of the plaintext the key will be repeated till the length of the plaintext.

In this algorithm a single key this plays the role of a generator key. i.e the key will be used for generating as many number of keys as equal to the length of the plain text. The key length can be varied and it should be divisible by two. To provide more security its better to choose the key length proportional to the message length. The key should be selected by either of two communicating entities and securely transferred to the other entity by means of a public key cryptosystems like RSA or Diffie Hellman.

The key generation is as follows. The key will be represented as bits and is made into two halves. The remaining keys will be generated by performing Left Circular Shift (LCS) and Right Circular Shift (RCS) alternatively on both halves of the key. For the first letter the key as such will be used and for the second character the first half of the key is shifted one bit left and the value is used as the key by keeping right half constant. For the third character the right half of the key left shifted one bit by keeping the left half as constant and that value is taken as the key. This process will be repeated till the end of the plain text. Once the key generation is completed the keys will be XOR ed with the plaintext to form the cipher text with respect to mod n. For decryption the cipher text will be XOR ed with the keystream will give the plain text. The key generations are as follows.

Consider the random key as K, it is divided into halves LH_i and RH_i. The key stream is given as

$k_1=LH_i \,||\, RH_i$
$k_2=LCS_1(LH_i) \,||\, RH_i$
$k_3=LH_{i+1} \,||\, LCS_1(RH_i)$
$k_4= LCS_1(LH_{i+1}) \,||\, RH_{i+1}$
$k_5= LH_{i+2} \,||\, LCS_1(RH_{i+1})$

Similarly the process will be continued till i=n-1, where n is the length of the plain text. The encryption and decryption process is given in eq. 7 and eq. 8 respectively.

$$C_i= P_i \text{ XOR } k_i \bmod 26 \quad (3)$$
$$P_i= C_i \text{ XOR } k_i \bmod 26 \quad (4)$$

3.2 Cryptanalysis of the New Algorithm

- The frequency analysis is not possible to perform in this algorithm because the characters are encrypted as one to many. i.e. a single character is mapped to many characters while performing encryption.
- Kasiski Test fails because the key in this algorithm is not repeating, i.e. the key is equal to the message size.

Advantage:

Though we are using many keys for encryption and decryption, the key exchange involves only one key. It is not necessary to use a single key for the encryption as used in one time pad, the key generation function can be used for using different keys for each character in the plain text.

4. Conclusion

Cryptanalysis of various classical encryption algorithms are done. The analysis shows that for small key space and plaintext all the algorithms are Vulnerable. To provide better security, it is advisable to choose large key space or go with more secure algorithm which may be complex to implement. The proposed algorithm could able to overcome the existing drawbacks of the classical encryption schemes.

5. References

1. Birkett J, Dent AW. Security models and proof strategies for plaintext aware encryption. Journal of Cryptology. 2014; 27(1):99–120.
2. Behrouz AF Debdeep M. Cryptography and Network Security. 2nd edition. Tata McGrawHill; 2007.
3. Bernard M. Network Security and Cryptograpgy. 1st edition. Cengage Learning; 2010.
4. Enes P. Probabilistic versus deterministic algebraic cryptanalysis - A performance comparison. IEEE Transactions on Information Theory. 2009; 55(11):5233–40.
5. Hung-Min S. Cryptanalysis of public key cryptosystem using generalized inverse of matrices. IEEE Communication Letters 2001; 5(2):61–3.
6. Rosen KH. Elementary Number Theory and Its Applications. 2nd Edition. Addison-Wesley; 1988.
7. Ramesh A, Suruliandi A. Performance Analysis of Encryption Algorithms for Information Security. International Conference on Circuits, Power and Computing Technologies 2013. 2013 March 20-21; Nagercoil, Tamilnadu, India; 2013. p. 840–4.
8. William S. Cryprography and network Security Principles and Practice. 5th edition. Prentice Hall; 2011.
9. Ganeshkumar K, Arivazhagan D. Generating a digital signature based on new cryptographic scheme for user authentication and security. Indian Journal of Science and Technology. 2014 Oct; 7(6):1–5.
10. Swapna BS, Sivanandam N. A survey on cryptography using optimization algorithms in WSNs. Indian Journal of Science and Technology. 2015 Feb; 8(3):216–21.