Cryptographic system based on Unicode

Cryptographic system based on Unicode

Noor D. AL-Shakarchy[1], Huda F. AL-Shahad[1] and Dhamyaa A. AL-Nasrawi[1]
[1]Computer Science Department, College of Science, University of Kerbala, Iraq

dh.alnasrawy@uokerbala.edu.iq

Abstract. During recent years, information security plays major roles with the wide use and development in communications networks. Stronger encryption and decryption represent the important demand in most applications. This strength can be achieved by data encryption and decryption algorithms which are very hard to crack and accomplish. The CIA security aims are Confidentiality, Integrity and Authenticity. In this paper, a novel cryptographic system was proposed based on principles of Unicode and crossover; This new combination add high level of security to proposed system. The English plaintext was converted to symbols in Unicode, key generated based on Arabic alphabet(also converted to Unicode), while the encryption and decryption method implemented depend on crossover principle by encryption and decryption keys K1,K2. Two parents (English plaintext, and encryption key) selected for mating using Average Crossover method (AC) which product one child only (Encrypted text) and vice versa in decryption method using decryption key. The security was achieved by confusion and diffusion concepts; variations and randomness in the encrypted text make the features of the original language are unclear; so the prediction of the plain text is difficult.
Keywords: Crossover, Cryptographic, Ciphertext, Key generation, Randomness, Unicode.

1. Introduction

Cryptography is the science and study of secret writing where by plaintext (or clear text) is transformed based on key in to ciphertext (cryptogram) by using an encipherment (encryption) process. The reverse process of encipherment of transforming ciphertext based on key into plaintext can be known as decipherment (decryption). The wide use of the internet which can be considered insecure network increase the demand to development of cryptography methods which can be able to store, transmit and deal with sensitive and important information[1]. Shannon presented the characteristics of good ciphers by:

- The amount of appropriate encryption and decryption work should be determined by the amount of secrecy needed.
- The complexity of the set of keys and enciphering algorithm should be free.
- The cipher system should be as simple implementation as possible.
- The error propagation in ciphering message should as less as possible so that if an error appear in early message this must not cause corruption of further information in the message.
- The ciphertext size should be no longer than the original message.

To design any encryption and decryption algorithm the Shannon's characteristics of good ciphers must be taken in account. In the other hand, improve the security level of encryption and decryption algorithm is major purpose to any design[2].

In this paper, new cryptograph system based of Unicode and crossover concepts was proposed. Many features were considered to make the system achieves confidentiality, integrity and authenticity such as randomness and variations. The contributions of this paper are shorten as below:

1. Create (char, int) mapping table for English alphabet, that mean give number for each character in English alphabet. In this step, int of b does not necessarily come after int of a, this called (Alphabet interleaving).

2. Convert English plaintext to Unicode plaintext, two English characters to one Unicode character, which can be hide the natural features of the original language, this step adds more robust against the statistical cryptanalysis.

3. Key management implemented based on Arabic alphabets, also converted to Unicode based on mapping table for Arabic alphabet. The generated key K1 used in encryption method, while the decryption key K2 which constructed from initial key K1, used for decryption method.

4. Used crossover principle in genetic algorithm, the resulted child represent the Unicode encrypted text.

The organization of this paper is: Principles of Unicode presented in section2, Section 3 introduces the related works. Section 4 show the methodology and GUI. Statistical cryptanalysis and results shown in section 5. Finally, conclusions in section 6.

2. Principles of Unicode

A character is the smallest composition of a text. Characters are traditionally represented by single byte values (allows for 256 characters) in most languages. When a system is used for a new language, the encoding has to be adapted to use that language 's characters[3,4].

Unicode is one Universal Code for every character no matter what the program, platform, or language. Unicode started out using 16-bit characters instead of 8-bit characters. That means 2^16 = 65,536 different values available, making it possible to represent many different characters from many different alphabets; an initial goal was to have[5].

"characters" is the basic elements of Unicode which called code points. Code points are specified by number, usually written in hexadecimal with the prefix "U+", such as U+006E "n" Latin small letter or U+0020 "space" . Each code point also has a short name, and quite a few other properties, specified in the Unicode Character Database. Codespace mean the set of all possible code points. The Unicode codespace consists of 1,114,112 code points which partition into 17 planes, from 0 to 16[6].

The proposed system concentrated in the first plane, Plane 0 "Basic Multilingual Plane", or BMP. All the characters needed for modern text in any script contains basically in BMP, including Latin, Cyrillic, Greek, Arabic, Japanese, Chinese, Korean, Devanagari (Indian), Hebrew, and many more. See roadmap of Basic Multilingual Plane in figure 1.

3. Related Works

Maram Balajee(Maram Balajee,2011) introduces a new method for cryptography by using UNICODE and colors. This work based on Private-key cryptography. The range of colors will be decided by the sender which will be assigned to 1,00,000 UNICODE characters. The color selected in such a way from three ranges of colors to generate the Unicode character range. The sample binding between character/symbol/digit, UNICODE and Color determined by create a dynamic mapping table. In this table the Unicode for each character of alphabet is determined then the corresponding color is determined. This table created before the encryption process.

The encryption process based of this dynamic mapping table. A color-chart of each page of text can be created. The drawback of this method can be detected by: The Unicode based in this method determined equaling of each character, This method used the predefined mapping to give the corresponding color for each UNICODE character. according to the predefined mapping[7].

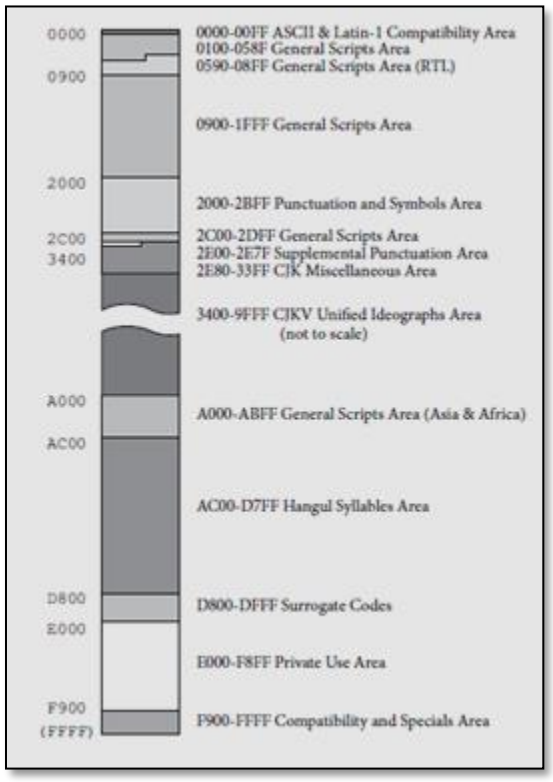

Figure 1. Basic Multilingual Plane Roadmap

A.Joseph and V.Sundaram (A.Joseph and V.Sundaram, 2012) presented an algorithm of "Data Encryption through Fibonacci Sequence and Unicode Characters". The plaintext (original) message are converted into the cipher text by searching each character in the message and interchanging it with character based on the Fibonacci number generated. Further, the cipher text is converted into the Unicode symbols[8].

In proposed system, create mapping table for English alphabet and for Arabic alphabet, then convert English plaintext to Unicode plaintext, key generation based on Arabic alphabets which converted to Unicode. The principle of crossover used to produce the Unicode encrypted text.

4. Methodology of Proposed Cryptography System

In this section, new cryptographic system proposed based on converting plaintext to Unicode, then use crossover in encryption and decryption process. the methodology of new cryptographic system is illustrated in detail; many steps must be implemented to encrypt and decrypt English plain text. Show figure 2.

The details of steps (that has been implemented in C#.net language). demonstrated in the following subsections.

4.1. Input English Plain Text

This is a first step in proposed system, the allowed plain text are English alphabet, English numbers, and some punctuations symbols. This step contains alphabet interleaving process, which create (char, int) mapping table for English alphabet. Table 1 show the mapping table for English alphabet in proposed system.

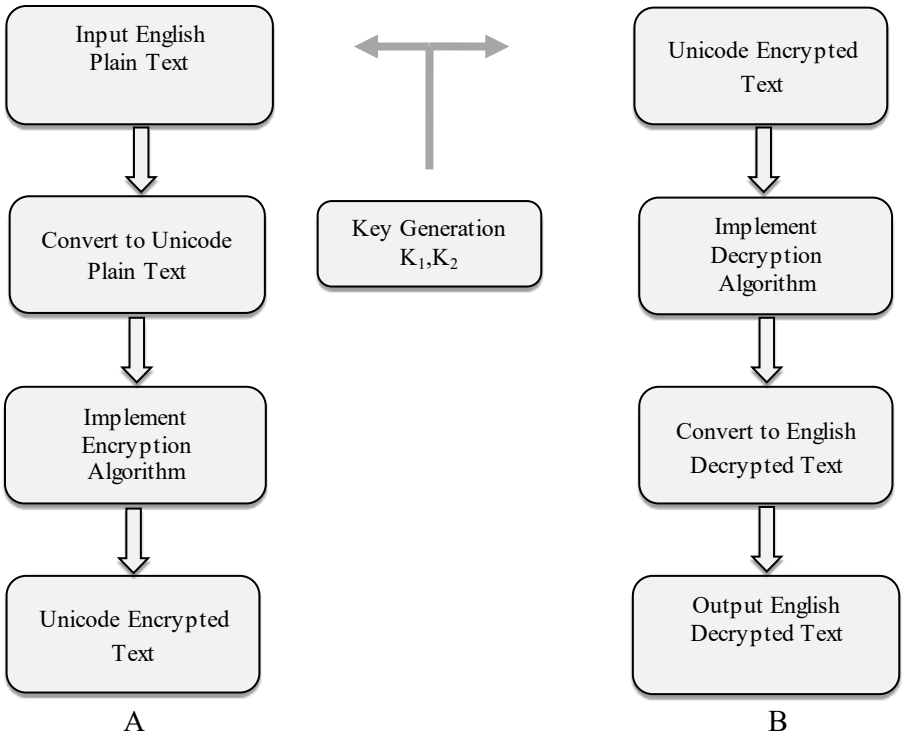

Figure 2. Steps of Proposed Cryptographic System, A: Encryption Process, B: Decryption Process

Table 1. Mapping Table of English Alphabet

char	*int*	*char*	*int*
a	38	v	49
b	82	w	76
c	60	x	35
d	44	y	32
e	13	z	56
f	64	?	66
g	68	.	54
h	81	,	83
i	59	$	78
j	61	space	30
k	79	:	72
l	42	0	33
m	80	1	73
n	47	2	52
o	18	3	23
p	20	4	21
q	36	5	70
r	26	6	75
s	69	7	55
t	15	8	12
u	63	9	67

Dictionary data structure in c# used to implement this step, see the following pesedo code that explains mapping process:

Mapping process
Inputs: English plain text
Output: get int for each char in English plain text
foreach char in English plain text: if char in dictionary: get int (value of char) else: do nothing.

4.2. Convert to Unicode Plain Text

The principle of Unicode was used to introduces more complexity since Unicode is committed not just to supporting texts in any single language, but also to letting multiple languages coexist within one text, which is the objective of this step.

The converting process explained in pseudo code as follow:

Convert to Unicode plain text process
Inputs: English plain text
Output: Unicode plain text
foreach *two char* in English plain text: - Get *int* (value of char) - Combine two *int* of two *char* together. - Get Unicode char of two *int* - Display the result

The details of this step explained in example below:

Let the English plain text: ***no pain no gain.***

1. By mapping table of English alphabet , get the *int* of each *char* are:

n	o	space	p	a	i	n	space	n	o	space	g	a	i	n	.
47	18	30	20	38	59	47	30	47	18	30	68	38	59	47	54

2. Combine two *int* together:

n	o	space	p	a	i	n	space	n	o	space	g	a	i	n	.
4718		3020		3859		4730		4718		3068		3859		4754	

3. Get Unicode and result

4718	3020	3859	4730	4718	3068	3859	4754
䜘	〠	㡙	䜰	䜘	と	㡙	䝔

In traditional converting to Unicode "n" = U+006E Latin small letter, while in proposed converting "n" combined with next letter to outcome Unicode as shown in above table "no" = U+4718, as well as "n" combined with another letter result new Unicode, example "n." = U+4754

4.3. Key generation

The process of generating key that used to encryption and decryption called Key generation, at first both keys are equal. In proposed system, this process implemented depending on Arabic alphabet by (abjad) order not on (hija'ai) order. The name (abjad) is based on the old Arabic alphabet's first four letters (أ, ب, ج, د). All Arabic alphabet collected in one string, followed by letters of (surat AL-Fath, aya 29 from Holy Quran) which contains all Arabic letters.

As described in the previous steps, Arabic alphabet has mapping table also, see table 2, then converted to Unicode in same as before. See the part of key generation process in the following example.

Table 2. Mapping Table of Arabic Alphabet

char	*int*	*char*	*int*
ا	16	ف	50
ب	19	ق	51
ت	24	ك	53
ث	25	ل	57
ج	27	م	58
ح	28	ن	62
خ	29	ه	65
د	31	و	71
ذ	34	ي	74
ر	37	ى	10
ز	39	ء	11
س	40	آ	14
ش	41	ا	17
ص	43	إ	77
ض	45	ؤ	85
ط	86	ة	22
ظ	87	ئ	84
ع	46	space	30
غ	48		

The details of key generation process that generate encryption key K1 explained in example below:
Let the part of Arabic alphabet in (abjad) order: أبجدهوزحطيكلمن

1. By mapping table of Arabic alphabet, get the int of each char are:

أ	ب	ج	د	ه	و	ز	ح	ط	ي	ك	ل	م	ن
16	19	27	31	65	71	39	28	86	74	53	57	58	62

2. Combine two int result these codes:

أ	ب	ج	د	ه	و	ز	ح	ط	ي	ك	ل	م	ن
1619		2731		6571		3928		8674		5357		5862	

3. Get Unicode and resulted key:

1619	2731	6571	3928	8674	5357	5862
১৩	✱	斀	烘	蚒	南	塢

As it's known, Unicode range from 0621 to 063A and from 0641 to 064A for the Arabic letters , while diacritics and images, range from 0600 to 06FF[3,9].
So, the letter "ج" as example = U+062C in traditional Unicode, , while in proposed converting "ج" combined with next letter such that "جد" = U+2731.

4.4. Implement Encryption Algorithm

This step focuses on using principle of crossover in genetic algorithm. Crossover is a genetic operator used to modify the chromosomes from one generation to the next. It works by taking more than one parent and producing a child from them.

In proposed system, crossover operator is the encryption algorithm, two parents represent Unicode plain text and Unicode encryption key, the produced child represent Unicode encrypted text. Average crossover method was used that reproduction one child only by average of the two parents, each gene

in a child is produced by averaging genes from both parents[10]. The equation of encoding and construction decryption key are done together, see equation of encoding in details where, Parent1 ,Parent2 ,Child represent Unicode plain text, Unicode encryption key (K1) and Unicode encrypted text respectively:

$$Child = (Parent1 + Parent2)/2 \quad \ldots(1)$$

$$\text{If } (Parent1+Parent2) \text{ is odd} \begin{cases} \lceil Child \rceil \ , K2 = K2+1 \\ \lfloor Child \rfloor \ , \text{no change on } K2 \end{cases} \quad \ldots(2)$$

We note that after the encryption process the decryption key K2 was constructed, the encryption process explained in pseudo code as follow:

Encryption process
Inputs: Unicode plain text, Unicode encryption key
Output: Unicode encrypted text
foreach *int* in Unicode plain text, *int* in Unicode encryption key K_1: - Compute average crossover from equations(1,2) - Replace new *int* (gene) to produced child. Get Unicode char of produced child Display the result

The details of implementation encryption process shown in below:

$Parent_1$ Unicode plain text	4718	3020	3859	4730	4718	3068	3859
	識	㉿	幌	祿	識	と	幌

Parent2 Unicode key K1	1619	2731	6571	3928	8674	5357	5862
	ᔕ	✱	斃	恟	蚖	南	塢

Produced Child Unicode encrypted text	3169	2876	5215	4329	6696	4213	4861
	ㄹㅅ	⁙	刱	織	暖	篤	鞜

4.5. Implement Decryption Algorithm

In the other side, the process is opposite, the decoding process implemented on Unicode encrypted text using Unicode decryption key K2 by the following equation:

$$Parent1 = (Child * 2) - Parent2 \quad \ldots(3)$$

Where, Parent1 ,Parent2 ,Child represent Unicode decrypted text, Unicode decryption key K2 and Unicode encrypted text respectively.

The decryption process explained in pseudo code as follow:

Decryption process
Inputs: Unicode encrypted text, Unicode decryption key
Output: Unicode decrypted text
foreach *int* in Unicode encrypted text, *int* in Unicode decryption key: - Retrieve Unicode plain text from equation(3) - Replace new *int* to Unicode decrypted text Get Unicode char of decrypted text Display the result

The details of decryption process explained below:

Unicode encrypted text	3169	2876	5215	4329	6696	4213	4861
	ㄹㅆ	⁙	劦	纔	暖	篙	輡

Unicode decryption key	1620	2732	6571	3928	8674	5358	5863
	ര	⁕	鼓	㤨	蚝	単	浧

Unicode decrypted text	4718	3020	3859	4730	4718	3068	3859
	譤	[illegible]	幌	徲	譤	と	幌

4.6. Convert to English Decrypted Text

This step is opposite of step 2, the Unicode number was split to produce two int according to mapping table of English alphabet.

The converting process shown in the following pseudo code:

Convert to English plain text process
Inputs: Unicode decrypted text
Output: English decrypted text
foreach *value* in Unicode decrypted text: - Split into two *int* - Get *char* of each *int* - Display the result

The details of this step explained as example below:

譤	[illegible]	幌	徲	譤	と	幌	獾
4718	3020	3859	4730	4718	3068	3859	4754

4718		3020		3859		4730		4718		3068		3859		4754	
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
n	o	space	p	a	i	n	space	n	o	space	g	a	i	n	.

47	18	30	20	38	59	47	30	47	18	30	68	38	59	47	54
n	o	space	p	a	i	n	space	n	o	space	g	a	i	n	.

Figures 3,4 and 5 show the GUI of proposed system with three examples.

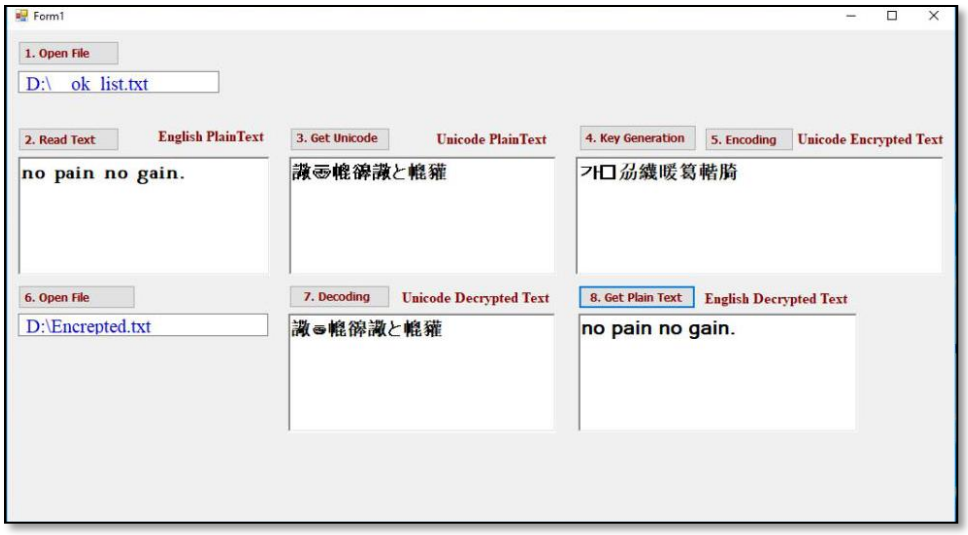

Figure 3. GUI of Proposed Cryptographic System

Now in another example:

Figure 4. GUI of Proposed Cryptographic System with Second Example

Figure 5. GUI of Proposed Cryptographic System with Third Example

5. Statistical Cryptanalysis and Results

All natural language have statistical characteristics, which means that each character in the alphabet has it’s own frequency in any text. Since these frequencies are so consistent, then an approximate probability can be attached to each letter. The most significant letter in a monoalphabetic cipher is equivalent for letter "e". However, English characters can be grouped in to five sets according to their frequencies, see table 3[11]:

Table 3 Frequencies of English Characters

Group	*Letter*	*Frequency*
1	e	9.614
2	t, a, o, i, n, s, h, r	6.855 – 4.532
3	d, l	3.219 – 3.047
4	c, u, m, w, f, g, y, p, b	2.106 – 1.129
5	v, k, j, x, q, z	0.741 – 0.056

The frequency of letters appearance satisfies the probability law: $\sum_{n=z}^{a} \mathbf{P}\ n = 1$. Cryptanalysis can be described as the art and science of breaking ciphertext. Therefore any cipher system can be measured by evaluating the security of cryptosystem. This measure concerns the computational effort required to break a cryptosystem. We might define a cryptosystem to be computationally secure if the best algorithm for breaking it requires at least N operations, where N is some specified, very large number so that it requires a large amount of computer time.

Now, in this section, we will evaluate the security level of the proposed system according to it's ability to pass the statistical analysis. Three experimental results were implemented with plaintext length 759, 239, 248 respectively, see figures 6,7, and 8.

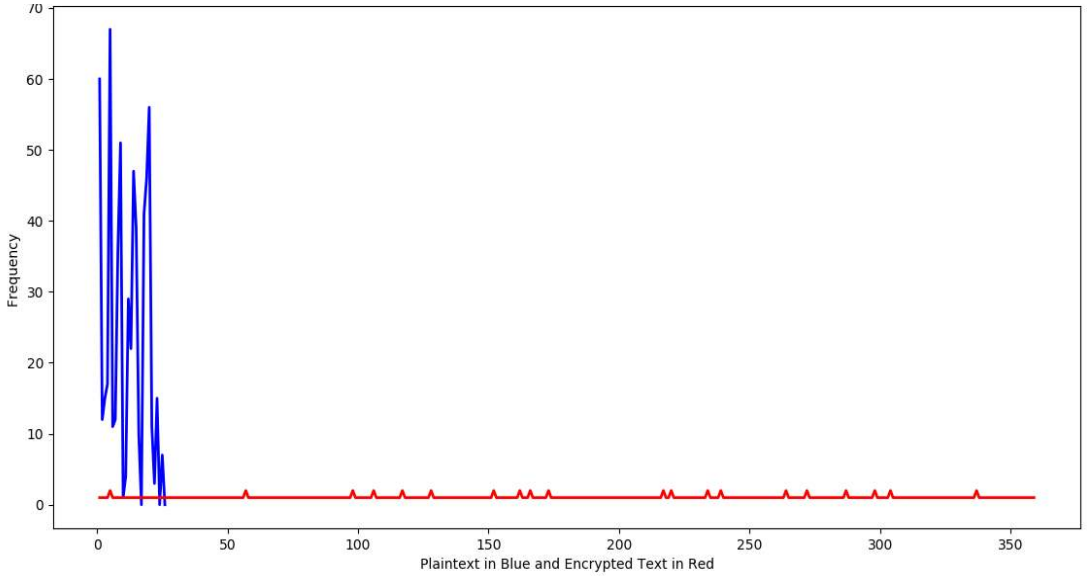

Figure 6. Histogram of Plaintext with Length 759 and Encrypted Text

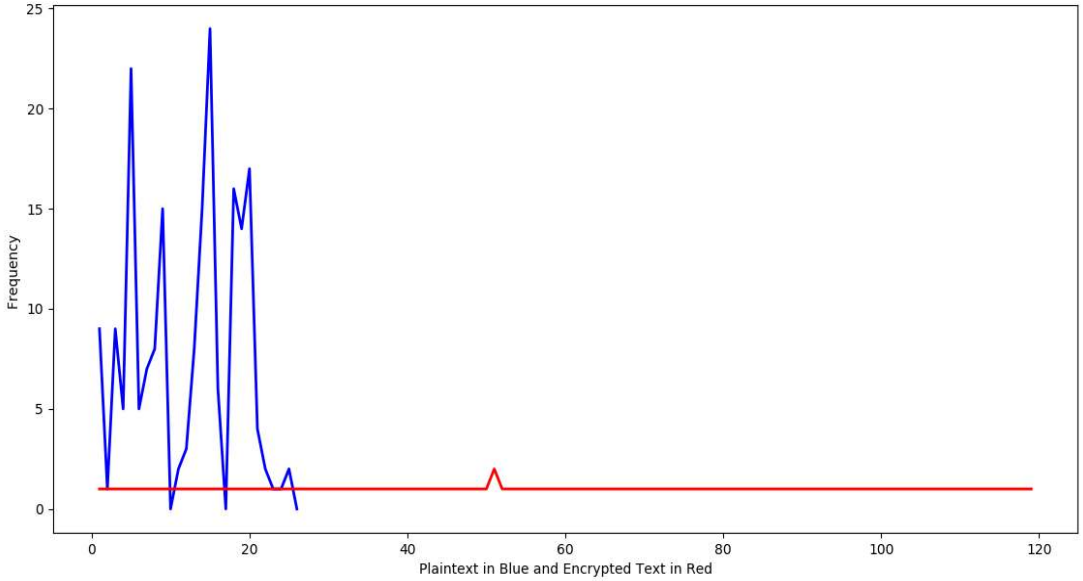

Figure 7. Histogram of Plaintext with Length 239 and Encrypted Text

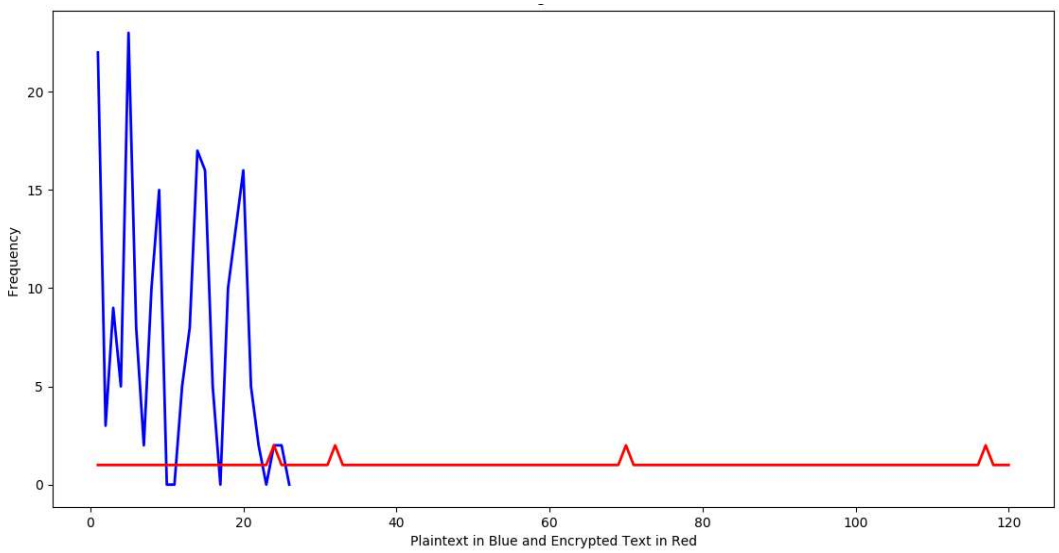

Figure 8. Histogram of Plaintext with Length 248 and Encrypted Text

As shown in figures, the alphabet of encrypted text was increased as well as no high frequency for each character. That mean the effort required to predict the plaintext, key or both from encrypted text is much more.

6. Conclusions

A novel method of cryptographic system was proposed based on combination of principles of Unicode and crossover. Sufficient amount of secrecy can be introduced by many steps, firstly in alphabet interleaving process which create mapping table for English alphabet used in plaintext and mapping table for Arabic alphabet used in key generation. Secondly in converting to Unicode process; this implemented on English plaintext and Key. These steps make the proposed system more robust against the statistical analysis. The statistical characteristics of plaintext message (natural language) scattered such that the attacker can't predict them. Thirdly in crossover process by mixing two parents and produce new child only.

The converting to Unicode process in proposed system provides:

- Generality and flexibility; it can be exceeds the limitation of alphabet; the encrypted text has more alphabet than plaintext.
- Randomness and variation; by convert two plaintext character to one Unicode character with very small frequency.
- Compression; presented in encrypted text. Plaintext with length N encrypted to encrypted text with length N/ 2. That makes it faster in transmission.

In addition, the proposed method is not needed especial hardware capability.

7. References

[1] Bruce Schneier , " Applied Cryptography Protocols, Algorithms, and Source Code in C ", Second Edition, John Wiley&Sons,Inc., 784 p., 1996.
[2] Henk C.A. van Tilborg, "Cryptology: A Professional Reference and Interactive Tutorial ", First Edition, Kluwer Academic Publishers Norwell, MA, USA, ,503 p., 1999.
[3] The Unicode Consortium, "The Unicode Standard", Available from: http://www.unicode.org.
[4] Yannis H. "Fonts & Encodings". Sebastopol, CA, USA: O'Reilly Media, Inc. 1040 p., 2007.
[5] Dhamyaa A. AL-Nasrawi et al., " Unicode Text Editor for Ancient Egyptian Hieroglyphs Writing System", *Egyptian Computer Science Journal* Vol. 38 No. 2 ,pp 48-55, May 2014.
[6] Jukka K. "Unicode Explained", USA: O'Reilly Media, Inc. 678 p., 2006.

[7] Maram Balajee, "UNICODE and Colors Integration tool for Encryption and Decryption", *International Journal on Computer Science and Engineering* (IJCSE), Vol. 3 No. 3 , pp 1197-1202, Mar 2011.
[8] A.Joseph and V.Sundaram, "Secured Communication through Fibonacci Numbers and Unicode Symbols", *International Journal of Scientific & Engineering Research*, Vol. 3, Issue 4, pp. 1-5, April-2012.
[9] Dhamyaa A. AL-Nasrawi, Ahmed F. Almukhtar, Wafaa S. AL-Baldawi, " From Arabic Alphabets to Two Dimension Shapes in Kufic Calligraphy Style Using Grid Board Catalog", Communications in Applied Sciences, Vol. 3, No. 2, pp 42-59, 2015.
[10] A.J. Umbarkar , P.D. Sheth, " Crossover Operators in Genetic Algorithms: A Review", *ICTACT JOURNAL ON SOFT COMPUTING*, Vol. 06, ISSUE 01, pp 1083 - 1092 , October 2015.
[11] William Stallings, “CRYPTOGRAPHY AND NETWORK SECURITY, PRINCIPLES AND PRACTICE”, FIFTH EDITION, Pearson Education, Inc., publishing as Prentice Hall, 2011