□

اقتراح خوارزمية
لتشفير النصوص مع إرسال مفتاح التشفير
ضمن النص المرسل

حنان رمضان مخور
قسم علوم الحاسبات
كلية العلوم / جامعة البصرة

اقتراح خوارزمية لتشفير النصوص مع إرسال مفتاح التشفير ضمن النص المرسل

حنان رمضان مخور
قسم علوم الحاسبات
كلية العلوم / جامعة البصرة
□

المستخلص:

مع التوسع السريع في علوم و شبكات الحاسوب، اصبح من السهل نقل البيانات الكبيرة عبر تلك الشبكات . هذا الامر يتطلب توفير الحماية والامنية اللازمة لنقل تلك البيانات بعيدا عن عيون المهاجمين او المخربين . التشفير هو احد الاساليب لحماية نقل البيانات في الشبكات المفتوحة .

في هذا البحث تم اقتراح خوارزمية لتشفير النص الصريح المدخل وكذلك توليد مفتاح التشفير بالاعتماد على خوارزمية لتوليده وارساله ضمن النص المشفر لأجل زيادة الحفاظ على المفتاح السري وضماناً لعدم تداوله عبر الشبكات المفتوحة .

الكلمات المفتاحية :التشفير، التشفير المتماثل، مفتاح التشفير، فك التشفير .

A Proposed Algorithm to Encrypt Text with the Encryption Key Included in the send text

Hanan ramahdan mokhour
University Of Basra
College Of Science

Abstract

With the rapid expansion in computer science and networks, it became easy to transfer large data over these networks. For this, it is necessary to provide protection and security crisis for the transfer of such data away from the eyes of the attackers or terrorists. Encryption is one method to protect the transport of data in open networks.

In this research an algorithm is proposed to encrypt the explicit text entry as well as generating the encryption key based on an algorithm for including generated key and sent it within the ciphertext for increasing and maintaining the secret key and not to be traded via open networks.

keywords: encryption, symmetric encryption, key encryption, decryption.

Introduction

There are many aspects to security and many applications, ranging from securing commerce and payments to private communications and protecting passwords. One essential aspect secure communications is that of cryptography .

Cryptography is the science of using mathematics to encrypt and decrypt data. Thus it enables to store sensitive information so that it cannot be read by anyone except the intended recipient. In data transfer and telecommunications, cryptography is necessary when communicating over any untrusted medium, which includes just about any network .

Cryptography, then, not only protects data from theft or alteration, but can also be used for user authentication. There are, in general, three types of cryptographic schemes typically used to accomplish these goals: secret key (or symmetric) cryptography, public-key (or asymmetric) cryptography, and hash functions. In all cases, the initial unencrypted data is referred to as plaintext. It is encrypted into ciphertext, which will in turn (usually) be decrypted into usable plaintext.[1]

Throughout history, however, there has been one central problem limiting widespread use of cryptography. That problem is key management. Conventional encryption (or symmetric) has benefits. It is very fast. It is especially useful for encrypting data . However, conventional encryption alone as a means for transmitting secure data can be quite expensive simply due to the difficulty of secure key distribution. For a sender and recipient to communicate securely using conventional encryption, they must agree upon a key and keep it secret between themselves.[2]

Related work

Lee et al. [3] proposed encryption algorithm to encrypt plaintext and do the reverse operation by applying basic computing operations such as inserting dummy symbols, rotating ,transpose shifting etc., to build the data in encryption algorithm.

Sahera. A.Saad [4] introduced algorithm to coding plaintext and do some operations on characters such as transformation ,inverse ,binaries ,substitute operations to generate encrypted file , and then doing the deciphering stage to obtain plaintext file.

Cryptography Principles

• Cryptosystem/Ciphers Rules

A cryptosystem is composed of a set [5]:

{Ee: e K } ---------- (1)

Where the key e ∈ K uniquely determines Ee acting upon plaintext to get ciphertext

Which consists of enciphering transformations. The corresponding set:

{Ee-1: e ∈ K} = {Dd : d ∈K} ---------- (2)

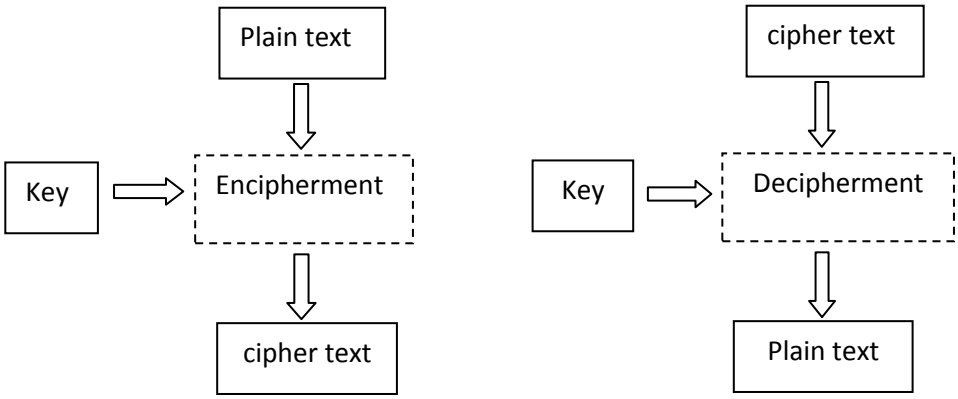
Which is uniquely determined by a given key d ∈ K, acts upon cipher text to get plaintext message units.

In other words, for each e ∈ K, there exists a unique d ∈K such that

Dd = Ee-1 ---------- (3)

so that Dd (Ee (m)) = m for all m ∈ M.

figure (1) illustrates the cipher process .



Fig(1) The software encipherment/decipherment processes

• Cryptography Goals

The cryptographic goals are privacy, integrity, authentication and nonrepudiation [7].

1. Privacy is the property of concealing the meaning of intent of message. In particular, it conceals it from undesired parts to an information transmission medium such as the Internet, wireless network link, cellular phone network, and the like.

2. Integrity is the property of ensuring correctness in the absence of any actively participating adversary .That sounds more complicated than it really is. Ensures that a message can be delivered from point A to point B without having the meaning (or content) of the original message change in the process. Integrity is limited to the

instances where adversaries are not actively trying to subvert the correctness of the delivery.

3. Authentication is the property of attributing an identity or representative of the integrity of a message. A classic example would be the wax seal applied to letters. The mark would typically be hard to forge at the time they were being used , and the presence of unbroken mark would imply the documents were authentic.

4. Nonrepudiation is the property of agreeing to adhere to an obligation. More specifically, it is the inability to refute responsibility. For example , if you take a pen and sign a (legal) contract, your signature is a nonrepudiation device. You cannot later disagree to the terms of the contract or refute ever taking part in the agreement.

• Key management

A key is the sequence that is used for the mathematical process of enciphering and deciphering information .Each pair of users shares a key for exchanging messages. Symmetric key management is the key management of cryptographic symmetric encryption keys. In a symmetric key algorithm the keys involved are identical for both encrypting and decrypting a message. Such keys must be chosen carefully, and distributed and stored securely. In any system there may be multiple keys for various purposes. Accordingly, key management is central to the successful and secure use of symmetric key algorithms.

The main characteristics of symmetric key management are:

Key generation, key exchange, key storage and key usage [8].

The benefit of symmetric key encryption is that it is fast, strong, and simple. This type of encryption allows to encrypt a large amount of data a short time ,but the problem how to provide security to send the key .If users are going to pass information in public medium such as internet then there must be same way to transfer the key and to prevent attackers from getting [5] .

In the proposed algorithm the key is to be drawn from plaintext and to be hidden inside the text .This ensures more security for sending the key and prevents its fall in hands of attackers.

- **Overview in some cipher systems**

1. Monoalphabetic substitution

In this system one alphabetic code is used for substitute .The system follow the is illustrated through the following example.

Plaintext: to be or not to be

Ciphertext: wr ehryq rw wreh

It can be seen that the letter (b) substituted the letter (e) by applying the relation:

F (a) = (a+k) mod n

Where (n) is the number of characters

(a) is the position of character

For the example:

F (b) = (1+3) mod 26 = 4=e

The number of shifting in character (k) is the key for this system, which is between o - 25 (English letters).

2. Transposed keyword mixed system

In this system a keyword is used. After deleting all repeated characters, a matrix is built, the number of its columns is equal to the length of keyword without repetition, the first row of matrix is the keyword and the alphabet characters are put in sequence in the other rows. For example:

Keyword: LONDON

L	O	N	D
A	B	C	E
F	G	H	I
J	K	M	P
Q	R	S	T
U	V	W	X
Y	Z		

The compensation has taken column after column as shown:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
L	A	F	J	Q	U	Y	O	B	G	K	R	V	Z	N	C	H	M	S	W	D	E	I	P	T	X

By using this system a good security is achieved because the degree of randomize is big.

Plaintext: to be or not to be

Ciphertext: sn au nh vns sn au

4- <u>Polyalphabetic substitution : Vigener cipher</u>

In this system there was many compensation doing on the text. In Vigener cipher choose a keyword by length D and put it under the plaintext , if the plaintext large than keyword so we repeat the keyword to cover all the character of message . the cipher operation is depending on relation :

$F(a)=(a+k_i) \bmod n$

Where n is the number of alphabet character, a is the sequence of letter that want to encrypt it , k_i is one of keyword character

where $d \geq i \geq 1$.

For example:

Keyword : BAND

Plaintext : RENA IS SANCE

Map the characters

								0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6

Repeat the keyword on plaintext

R ENA I SSANCE

BANDBANDBAN

The cipher operation doing by applying the relation above :

F (R) =(17+1)=18 mod 26 = S

F(E) = (4+0) = 4 mod 26 = E

and so on , then the cipher text is

Plaintext : RENAISSANCE

Ciphertext : SEADJSFDOCR

Proposed algorithm

In research will use Vigener system to encrypt files , and doing some operations to increase the degree of security and to give challenge to the algorithm .

1- Read plaintext file :

In this step read plaintext file as text file.

2- Delete spaces from the file to appear as block of characters without spaces

3- Generate the key (key generation algorithm) :

In this step generate key from plaintext by depending on key length and file size that had given in the beginning , choose the length of key is 4 character (you are free in the choice) . The algorithm as follow:

a. Take file size and divided it on key length the result of this operation is the first position of key take the value of this position and consider it the basic to compute the other positions of the key .

b. After h determine the first position examine it if it is even or odd.

c. If the position is an even number evaluate the next position by the equation:

Next_pose.=first_pose./4 + keylength – i(loop variable)

d. If the position is odd number , evaluate the next position by equation

Next_pos.=first_pose.*7 mod filesize

e. Take the character or symbols of these positions from plaintext , and cutting it from the file, so the file will reduce by length of key .

4- Transfer symbols to values

In this step will convert the symbols of the file into bytes , preparing it to first encryption .

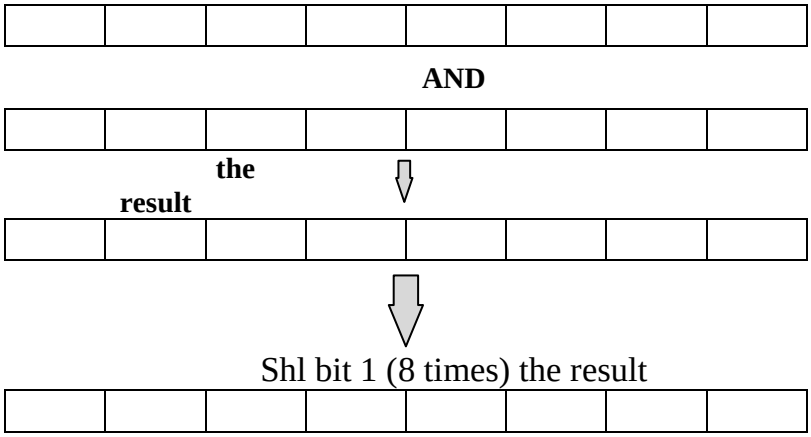
5- Vigener cipher operation

In this stage had got the ASCII code for symbols of the file the result of this step is encrypted symbols text .

6- Generate binary values

After doing first encryption generat binary values for cipher characters this operation do by taking the byte(value) and doing AND operation with $80(in hex) and shifting the result 1 bit to the left doing this operation 8 times , at the end will get binary value.

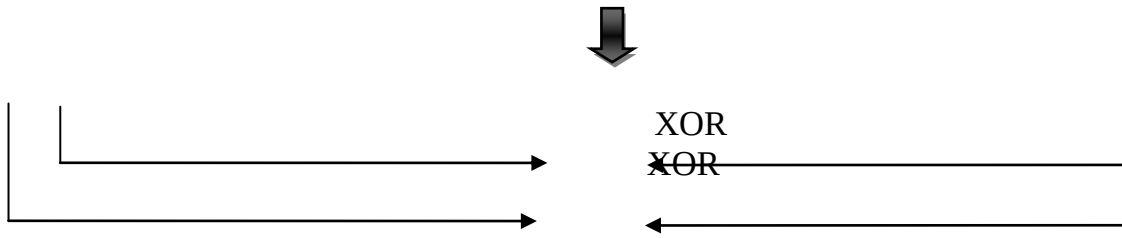
Like this example: byte and $80 ⟹ if the byte is the number 7



7- applying xor operations

In this step dividing cipher text file into two parts (file size/2) and applying the **xor** operation between these two parts by taking first character of first part with last character of second part and put the result in the first position ,and take second character with before last character and put the result in the second position and so on , by this have half of file encoding by **xor** operation .

For example :



A xor Z ⟹ byte the result is overlying in position A

B xor Y ⟹ byte the result is overlying in position B

8- Applying exchanging (switching) operation

From previous step have half file encoding by xor operation . In this step will take the first 4 bits from first character (first byte)and exchanging it with last 4 bits of last character . doing that for whole file , by switching have obtain different bytes from origin bytes .

can illustrate the operation as :

Take the number (111) and the number (C) in hexa system :

(111) 0 1 1 **after** 1 1 0 0 (2

0 1 1 1 1 **switching** 1 1 1 1 07)

(c) 0 0 0 0 0 0 0 0 (6

0 1 1 0 0 ⟹ 0 1 1 0)

9- Transfer binary value

From previous steps have got on binary values for encrypted characters , in this step will transfer these values into bytes and after that convert it into Ascii codes .

10- Return the key

This is the final step of algorithm , have an encryption file after gain on cipher text . By doing all the previous operations will return the key characters to its positions that have get it in start , in this way will hide the key inside the encrypted file in order to transmit it within the file.

The Decipher algorithm

Will describe the decryption that it is the obesity steps of encryption , the algorithm as follows:

1- Read the cipher text file .

2- Cutting or retrieve the key characters (cipher key) from the file , pursuant to its positions inside the file . Depending on key length and file size that generate key from these two factors .

3- Convert the file from ASCII code into bytes values.

now going to rewind the bytes value to its original , by dismating bytes in exchanging operation to its original .

4- In this step transfer the byte values from pervious step into binary values .

5- after having data from previous step will dismating the values , in this step apply xor operation between first byte(the result in cipher stage) and last byte of file to obtain on origin byte.

6- would get file of bytes from last step will convert it into ASSCI code , so have cipher text by Viginer cipher.

7- In this step apply the key that derive it from beginning on cipher text , this done by running key character with all file character (after transmit character into bytes) this step decrypt the text and have original text but without key character .

8- return the key character to its positions within the file. By this way will get an full origin characters file.

❖ **The experimental results**

display for all the results to ciphering as deciphering algorithm

First stage:: encryption stage

have the plain text file :

No.	Name	Address	No. account
101	Jones Tol Co	Chicago , Il	60605.
102	Hal Coputers, Inc	Armonic , Ny	10504.
103	Going System Group	Seattle, Wa	98124.
104	Toh Steel Co	Pittcburgh , Pa	15213.
105	Cipher System , Inc	Arlington , va	22209.
106	G & O Co , Inc	Huston , Tx	77002.
107	lsi Co , Inc	New Haven , Ct	07733.
108	I/O Devices Group,	Holmodel , Nj	06520.
109	Crt Inc ,	Fresno , Ca	63710.
110	Crypto System , Ltd	Rockville , Md	20852 .

1- The file without spaces :

```
101jonestoolcochicago,il60605.102halcoputers,incarmonic,ny10504.103g
oupseattle,wa98124.104tohsteelcopittcburgh,pa15213.105ciphersystem,i
va22209.106g&oco,inchuston,tx77002.107lsico,incnewhaven,ct07733.108i
up,hoomodel,nj06520.109crtinc,fresno,ca63710.110cryptosystem,ltdrock
2.
```

2- Generate the key from the file depending on key generation algorithm as follow :

```
the position   1 81    the value       o
the position   2 245    the value       o
the position   3 62    the value       0
the position   4 16    the value       h
```

By determine the key length is 4 keys then the encrypted key (oo0h)

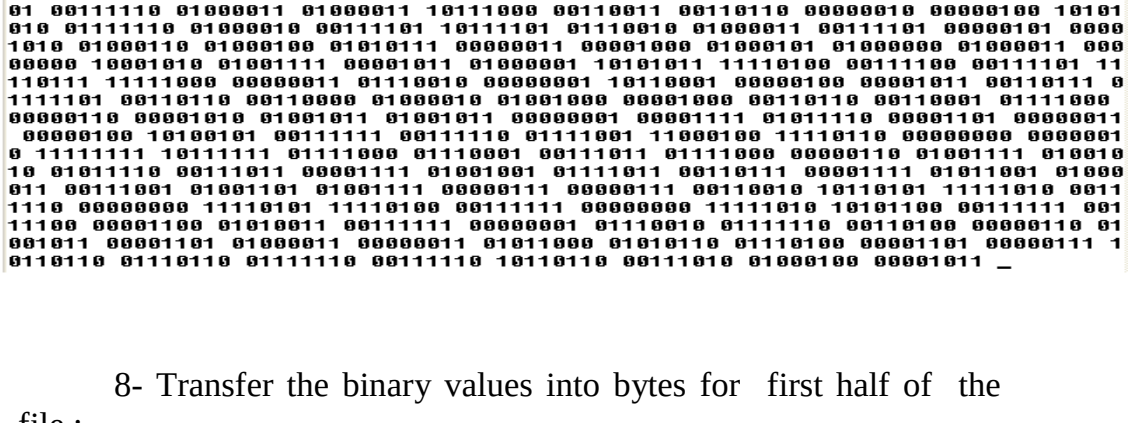
3- The file after cutting the key from it , the file size reduced by key length as follows:

```
101jonestoolcocicago,il60605.102halcoputers,incarmonic,ny1054.103go
seattle,wa98124.104tohsteelcopittcburgh,pa15213.105ciphersystem,inc
2209.106g&oco,inchuston,tx77002.107lsico,incnewhaven,ct07733.108i/o
homodel,nj06520.109crtinc,fresno,ca63710.110cryptosystem,ltdrockvil
```

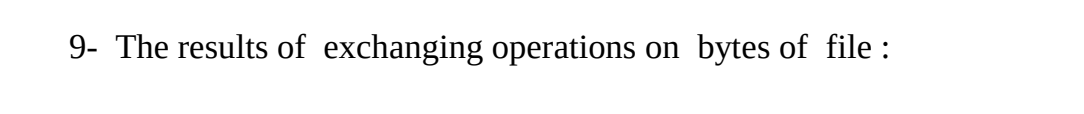
```
49484910611111010111511611111110899111991059997103111441051085448544
04971089911111211711610111411544105110999711410911111010599441101214
85110311110511010311512111511610110910311411711211510197116116108101
95052464948521161111041151161011011089911111210511611699981171141031
35049514649485399105112104101114115121115116101109441051109997114108
11111044118975050504857464948541033811199111441051109910411711511611
55554848504649485510811510599111441051109911010111910497118101110449
51464948561054711110010111810599101115103114111117711244104111109111
```

```
12 145 220 227 219 149 148 230 208 105 160 160 161 100 150 160 159 10
15 163 220 212 212 156 203 222 223 153 220 227 210 146 221 225 214 15
08 97 157 161 160 99 150 160 159 101 203 216 223 152 205 225 226 169
 157 148 216 221 147 201 225 219 153 214 214 227 159 214 155 229 145
 96 161 157 160 96 158 214 149 159 203 222 155 153 214 210 215 165 21
58 148 227 231 103 159 159 159 98 150 160 159 103 212 226 216 147 215
8 203 221 212 167 208 208 229 149 214 155 210 164 152 166 166 99 155
160 216 158 159 204 212 229 153 203 212 226 151 218 222 228 160 148 2
215 211 212 156 148 221 217 96 158 164 161 96 150 160 159 105 203 225
4 210 155 150 218 212 226 158 215 155 210 145 158 162 166 97 152 157
210 225 169 216 227 222 163 225 226 227 149 213 155 219 164 204 225 2
229 216 156 212 212 155 157 204 161 159 104 157 161 157_
```

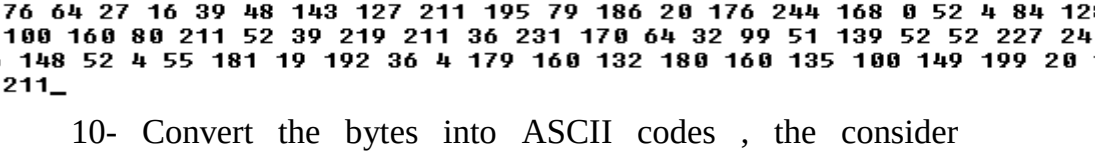
7- The results of XOR operation as binary values for first half of the file as follows:



8- Transfer the binary values into bytes for first half of the file :

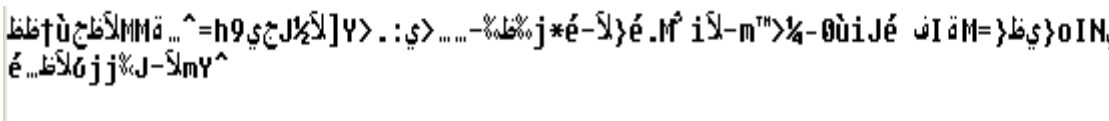


9- The results of exchanging operations on bytes of file :



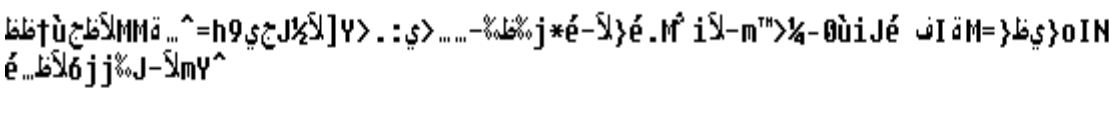
10- Convert the bytes into ASCII codes , the consider encrypted file as follows :

11- The encrypted file after adding the cipher key on original positions in the file , the values and position of key illustrate in step(2) , the result as follows:



Second part :: decryption stage

1- Read the encryption file :

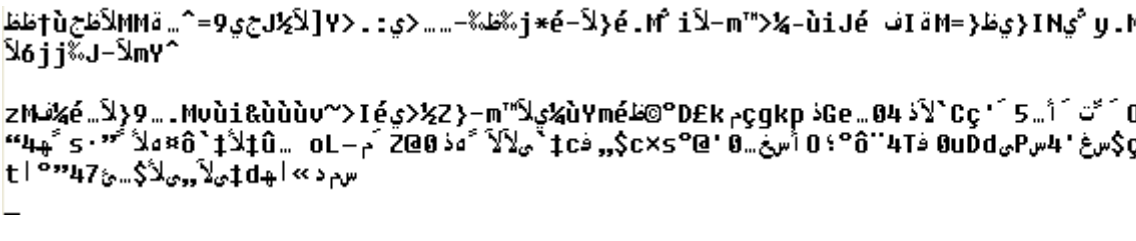


2- Retrieve the cipher key from the file depending on algorithm of key generation after determine its length , position and values :

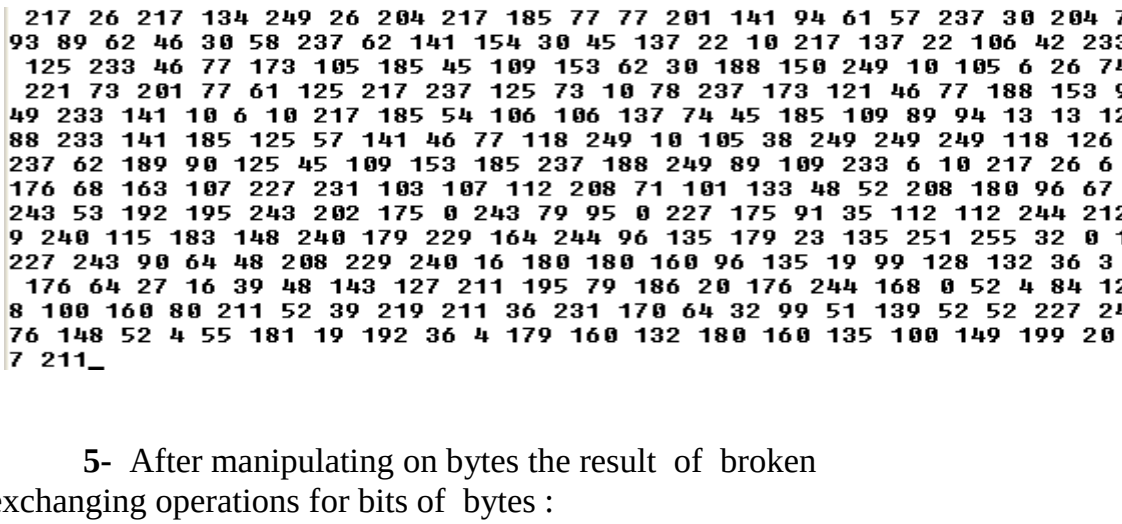
```
the position  1 81   the value     o
the position  2 245   the value     o
the position  3 62   the value     0
the position  4 16   the value     h
```

The ordinary of the key was (oo0h) .

3- The file after cutting the cipher key , the file reduced by key length :

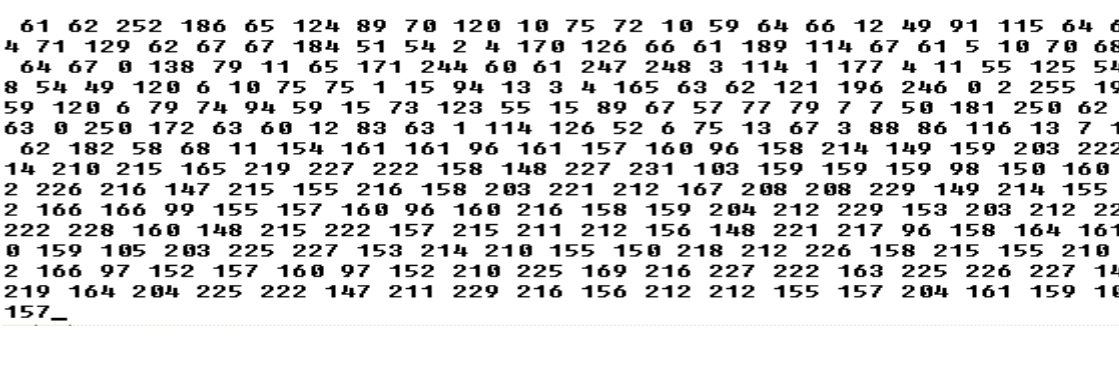


4- Transfer the ASCII symbols into bytes as follow :



```
 217 26 217 134 249 26 204 217 185 77 77 201 141 94 61 57 237 30 204 7
93 89 62 46 30 58 237 62 141 154 30 45 137 22 10 217 137 22 106 42 233
 125 233 46 77 173 105 185 45 109 153 62 30 188 150 249 10 105 6 26 7
 221 73 201 77 61 125 217 237 125 73 10 78 237 173 121 46 77 188 153 9
49 233 141 10 6 10 217 185 54 106 106 137 74 45 185 109 89 94 13 13 12
88 233 141 185 125 57 141 46 77 118 249 10 105 38 249 249 249 118 126
237 62 189 90 125 45 109 153 185 237 188 249 89 109 233 6 10 217 26 6
176 68 163 107 227 231 103 107 112 208 71 101 133 48 52 208 180 96 67
243 53 192 195 243 202 175 0 243 79 95 0 227 175 91 35 112 112 244 212
9 240 115 183 148 240 179 229 164 244 96 135 179 23 135 251 255 32 0 1
227 243 90 64 48 208 229 240 16 180 180 160 96 135 19 99 128 132 36 3
 176 64 27 16 39 48 143 127 211 195 79 186 20 176 244 168 0 52 4 84 12
8 100 160 80 211 52 39 219 211 36 231 170 64 32 99 51 139 52 52 227 24
76 148 52 4 55 181 19 192 36 4 179 160 132 180 160 135 100 149 199 20
7 211_
```

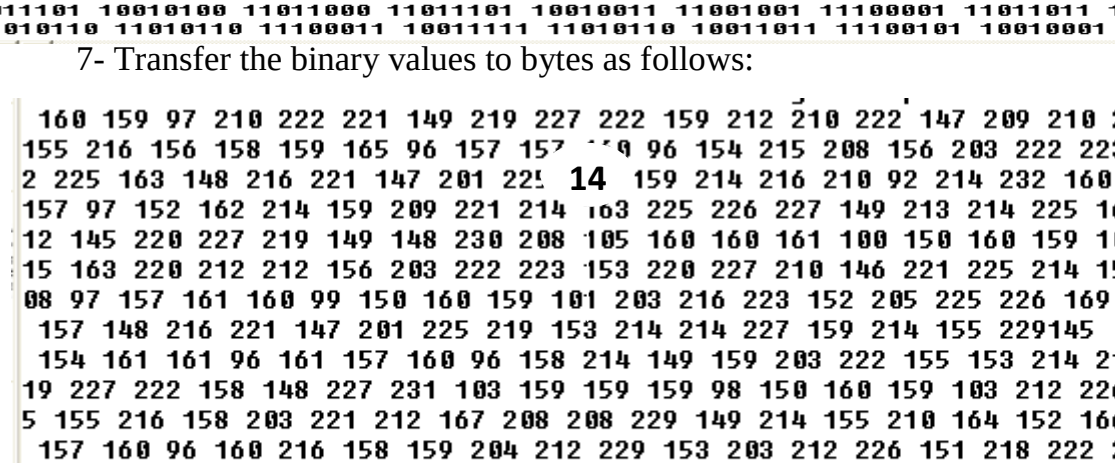
5- After manipulating on bytes the result of broken exchanging operations for bits of bytes :



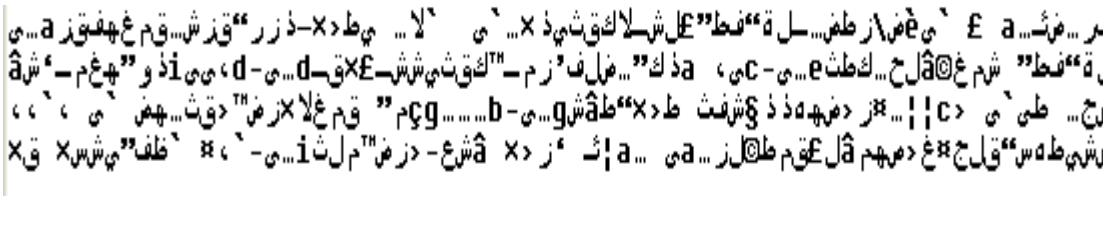
```
 61 62 252 186 65 124 89 70 120 10 75 72 10 59 64 66 12 49 91 115 64 6
4 71 129 62 67 67 184 51 54 2 4 170 126 66 61 189 114 67 61 5 10 70 68
 64 67 0 138 79 11 65 171 244 60 61 247 248 3 114 1 177 4 11 55 125 54
8 54 49 120 6 10 75 75 1 15 94 13 3 4 165 63 62 121 196 246 0 2 255 19
59 120 6 79 74 94 59 15 73 123 55 15 89 67 57 77 79 7 7 50 181 250 62
63 0 250 172 63 60 12 83 63 1 114 126 52 6 75 13 67 3 88 86 116 13 7 1
 62 182 58 68 11 154 161 161 96 161 157 160 96 158 214 149 159 203 222
14 210 215 165 219 227 222 158 148 227 231 103 159 159 159 98 150 160
2 226 216 147 215 155 216 158 203 221 212 167 208 208 229 149 214 155
2 166 166 99 155 157 160 96 160 216 158 159 204 212 229 153 203 212 22
222 228 160 148 215 222 157 215 211 212 156 148 221 217 96 158 164 161
0 159 105 203 225 227 153 214 210 155 150 218 212 226 158 215 155 210
2 166 97 152 157 160 97 152 210 225 169 216 227 222 163 225 226 227 14
219 164 204 225 222 147 211 229 216 156 212 212 155 157 204 161 159 16
157_
```

6- The result of dismating the XOR operations as binary values for first part of the file :

7- Transfer the binary values to bytes as follows:



8- Returning encryption file



9- Rewind the encrypt file to original file without key as follows :

```
101jonestoolcocicago,il60605.102halcoputers,incarmonic,ny1054.103goin
seattle,wa98124.104tohsteelcopittcburgh,pa15213.105ciphersystem,incar
2209.106g&oco,inchuston,tx77002.107lsico,incnewhaven,ct07733.108i/ode
homodel,nj06520.109crtinc,fresno,ca63710.110cryptosystem,ltdrockville
```

10- the original file after adding the cipher key as follows :

```
101jonestoolcochicago,il60605.102halcoputers,incarmonic,ny10504.103g
oupseattle,wa98124.104tohsteelcopittcburgh,pa15213.105ciphersystem,i
va22209.106g&oco,inchuston,tx77002.107lsico,incnewhaven,ct07733.108i
up,hoomodel,nj06520.109crtinc,fresno,ca63710.110cryptosystem,ltdrock
2.
```

❖ **Conclusion**

There are many applications for which symmetric encryption is the best choice, providing high security that is both efficient and most effective.

The experiments show the ability of proposed algorithm to encrypt the plaintext

By using secret key extract from text , depending on key length choice ,everywhere the key is long (not more than 128 bit) have more secret and the time for decrypted the text is long too.

Then the method satisfied the cryptography goals as

- Privacy : by encryption the original text and hiding the key within the text.
- Nonrepudiation : by converting the text value into ASCII code and any fraud on the text file causes that the message will not be decrypted correctly.

- Authentication : by using the secret key to encrypt text and decrypt it . with same key exactly.
- Integrity : by passing the text values in multistep encryption operations , the intruder will be unable to forge the message.

References

1- Lee . W , Chen . T , and Chieh Lee . C , "Improvement of an encryption scheme for binary images," *Pakistan Journal of* Information *and Technology*. Vol. 2, 2003 . http://www.ansinet.org/

2- Stallings, William., " Cryptography and Network Security" , Prentice Hall , 1999.

3- Lee .Tsang yean , Lee .Hueg ming , Wu .Homer , and Su .Jin shieh ," Data transmition encryption in network security ", World Scientific and Engineering Academy and Society (WSEAS) , U.S.A , 2006 .

4- Sahera A .Saad ,"Depending character and binary changing for information coding " , Scientific Journal of Thi_qar University , Iraq , 2006.

5- Mollin .Richard A. ," An introduction to cryptography",2nd ed .Taylor & Francis Group LLC, U.S.A , 2007.

6- konheim .Alan G. , "Computer security and cryptography" , John Wiley & Sons , Canada , 2007.

7- Tom St Denis , " Cryptography for developers " , Syngress publishing . U.S.A , 2007.

8- Levente Buttyán , and István Vajda , " Cryptography and its applications" , Typotex , ISBN 963-9548-13-8 , university of technical , Budapest , hungary , 2004 .