Review Article

ACCENTS Transactions on Information Security, Vol 2(8)
ISSN (Online): 2455-7196
http://dx.doi.org/10.19101/TIS.2017.28002

New secure and reliable polygraphic cryptosystem

Swapnil Paliwal[*]
School of Information Technology, VIT University, Vellore, Tamil Nadu, India

Abstract

An effective probabilistic polygraphic cryptosystem is developed, which is capable of performing compression as well as expansion encryption. There are two set of keys which are used, key1 is developed at the time of encryption whereas key2(is a matrix which can at most contain M X T keys in it) is used to change the arrangement(as it is multiplied with Alice's private arrangement) hence a secure exchange system is established, again we continuously shift from one polygraphic arrangement to the other while encrypting the same message after fixed number of encryption's then this will secure the cryptosystem from various attacks, and make it more compact and secure. Again using Diffie and Hellman key exchange for exchanging keys which are used to produce a unique arrangement. The key1 is generated in such a fashion that it displays as of how many fake characters are generated before every cipher character or key number and what operation is performed on those.

Keywords

Polygraphic cryptosystem, Keys, Pen-paper based system, Diffie and Hellman key exchange.

1.Introduction

Polygraphic ciphers such as playfair cipher is not robust since the arrangement can be exposed by frequency analysis, Here we develop a scheme such that it shifts from one arrangement to other while enciphering this will confuse or frustrate cryptanalysis and this will secure the system [1-3]. We have a method which can expand a character to such an extent that the net result of ciphertext tends to a very big number; again we have a compression algorithm which can compress a relatively big plaintext to a single character[4]. To prevent cryptanalysis we perform confusion and diffusion on every ciphertext character and key numbers. Again Alice and Bob can conclude to a same arrangement even while communicating via an insecure channel.

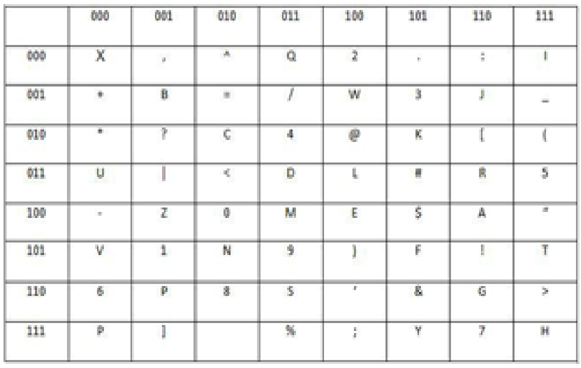
2.Description of the cryptosystem

This cryptosystem is based on compression and expansion encryption. There are two keys which are used in this system; the decryption key (which we call key 2) is generated at the time of encryption which will be used later for decryption purpose. First the polygraphic arrangement is solely generated by Alice, who performs function on her arrangement with the key (which we call key1 or arrangement equalizing key) generated using. Diffie and Hellman key exchange method (f (A; k1)) this key is a matrix of M X M dimension same as that of the arrangement. Now Alice sends here matrix S to Bob via an insecure channel, now S is known publically, since only Alice and Bob know the arrangement of k1 matrix thus now Bob can know the arrangement of Alice and can write secret messages, Although arrangements will be changed from time to time as to sustain higher security of this method (Although the method remains pretty secure because of the encryption schemes, which are discussed in the next section). Confusion and diffusion methodology is also used as to confuse or frustrate cryptanalyst who use frequency or thwart analysis for decrypting or predicting the text. The arrangement is similar to that of k-map (Veitch diagram) and each square holds a character or a number which has a particular value assigned as per the arrangement, the location of row and column is reproduced in the key (which is used for decryption) to retain the original character/'s used for encryption. The compression encryption focuses on reducing the cipher text length, whereas the expansion encryption focuses on doubling the length of the cipher text with respect to the source text. Since it is probabilistic scheme, thus these expansion and compression can be performed; Compression is usually accompanied by expansion encryption.

[*] Author for correspondence

3.Arrangement

The cryptosystem uses a diagram which is a k-map and characters occupy the location in the map and the rules which were discussed above are implemented and ciphers are created. We are using a completely random arrangement here, but in general it is expected that the arrangement follows an order i.e it should be as follows
00,001,010,011,100,101,110,111.
Still we assume that our arrangement is proper, and we use this arrangement only to demonstrate several examples. This is our general arrangement which is assumed as Alices private arrangement and is used throughout for encryption and decryption purpose. Again we discuss it later that how to parties can conclude to a same arrangement through an insecure channel, and if needed the arrangement can be changed which is shown in *Figure 1*.



	000	001	010	011	100	101	110	111
000	X	,	^	Q	2	.	:	!
001	+	B	=	/	W	3	J	_
010	*	?	C	4	@	K	[	{
011	U	|	<	D	L	#	R	5
100	-	Z	0	M	E	$	A	"
101	V	1	N	9	)	F	!	T
110	6	P	8	S	'	&	G	>
111	P	]		%	;	Y	7	H

Figure 1 Sample arrangement

4.Encrypting/decrypting scheme

The arrangement of this system is of M X M dimension and first the characters are randomly arranged, and then again after certain fixed encryption one can either decide to change the key or change the arrangement. Let us discuss about the rules associated with the encryption and later let us discuss as of how Alice or Bob equalize there arrangements. Now in encryption, the most important thing is that both Alice and Bob who are trying to communicate secretly through an insecure channel. Thus in order to make sure that the arrangement is shared secretly, Alice sends her f (A; k1) =S to Bob via the same insecure channel, now Bob performs his operation i.e f(S; k1) which enables Bob to know the arrangement of Alice. Once the arrangement is known to Both Alice and Bob, they can begin their conversation through the insecure channel. There are two encryption scheme used expansion and compression thus one can perform either expansion encryption or compression encryption (although compression expression solely cannot be performed.) but we perform compression expansion simultaneously for obtaining our desired result. This is a polygraphic probabilistic encryption scheme. Thus if compression with proper help of expansion encryption(if needed) is performed then there is a possibility that we can only have one character as the net ciphertext output , Either way combination of these two is more highlighted and discussed in this paper.

The key is generated at the time of decryption(for instance if we are to say that the character lies in 4th column and 4th row and 4 in the arrangement of 8X8 is at the location of 100110 then we write the key as 100110100110).

A. Compression encryption

Compression encryption scheme cannot work by itself, since it focuses on reducing the size of cipher text by half while passing through different encryption layers, since it highly probable that at some layer L the number of characters in that layer will be odd L =2n + 1 where n 2 W . The problem does not end here this method will work if there are no repeating characters forming up pairs together this is now a major drawback of this method and is highly advisable that this method must be used with expansion now let us establish rules of encryption which two parties must follow while encrypting texts, There are 3 cases involved here which are Characters in different row and column Characters in same row different column Characters in same column different row, let us discuss each of the above cases.

1) Characters in different row and column: This is the easiest encryption case compared to others. So for encrypting, Let the first plaintext character lie in the ith column and jth row of our arrangement and let our second plaintext character lie in the kth column and lth row then the cipher character will lie in kth column and jth row of our arrangement and key will be the column (i) in which the first character lies and row (l) in which the second character lies. Thus $[P\ 1]_{(i;j)}[P\ 2]_{(k;l)} = [C]_{(k;j)}$ and key will be key$_{decryption}$ = il.
Sub subsection characters in same row different column Both P1 and P2 lie in the same row j and different columns i.e i and k respectively, thus cipher characters will lie in (k+1)th column and jth row. Thus the cipher character lies in $[P\ 1]_{(i;j)}[P\ 2]_{(k;j)} = [C]_{(;j)}$ where = (k + 1)modM 2 and M is number of columns, Thus the key will be written in the way which was dicussed in the above method, thus
^{key}decryption $^{= il.}$

2) Characters in same column different row: Here characters lie in the same column and The encrypted character is[P 1](i;j)[P 2](i;k) = [C](i;) where =(j + x)mod M= (k + 1)modM in this case the M is number of rows but still it ought to make any difference as the number of rows equal number of columns, New rule must be established in order to write the decryption key if it is written in the same fashion the other keys were written then it will be tough to decipher characters since there will be multiple possibilities (for instance if $key_{decryption}$ = il then the character lies in ith column and lth row but if two characters are in the same column for instance they lie in 2nd column 3rd row and 2nd column 4th row then the key will correspond to 2nd column and 4th row but the cipher character actually lies in 2nd column 5th row) thus in order to avoid that we write the key in reverse fashion i.e row in which the first character lies and the column in which the second character lies. But the user who is decrypting is actually unaware that whether the character were in the same column thus we write a number which is not used in the arrangement for instance if the arrangement is of order 8X8 then we will use 9 in the key before writing the row in which the first character lies and column in which the second(similarly if arrangement is 9X9 then we use 10 if it is 10X10 we use 11 and so on), thus $key_{decryption}$ = li where is the unique number which is used to represent the change. Let us take an example to understand how compression encryption and decryption works,

Example

let us take the above presented arrangement for encrypting. Let our plaintext which is represented by P_T be, P_T =SWAPNIL.

The pairs formed are SW AP NI L. since we are encrypting the original text once, thus we call this as cipher text 1, and is represented by C_{T1}. Thus our cipher text is ^{0} T 2 (the key at this stage is 31672040).

Since this scheme is based on probabilistic encryption scheme thus we can proceed and generate second cipher layer till the net output of some cipher layer C_T reaches one character if desired. now let us generate C_{T2} we get 6) (the key at this stage is 4470) and C_{T3} is ^{0} (the key at this stage is 05) now the final key after k is 31672040447005 but since we write the key in form of the location of that number in the arrangement thus the key becomes k= 0011011010011110001101110001001000 1 1011010011010011010110111100011100011 01010

Since the final cipher text was just ^{0} and thus this must be expanded first, thus we use last key i.e 100011010101 to expand this when expanded it will point to the location of 0th column and 5th row thus now we know that because of a character which is present in 0th row and another character which is present in 5th row we get this while decrypting we find only one unique pair which satisfies this, which is 6).

B. Expansion encryption

In expansion encryption method, characters double by every layer for instance if a layer currently has 4 characters in it after expansion it will have 8 characters and again after encryption of this layer it will have 16 characters and so on thus, every succeeding layer will have characters equal to 2 in them, where is the elements in the current layer. This is the method which overcomes the drawback of compression encryption (although this method also cannot be used as is, because it will give cryptanalyst an idea about the arrangement). This is how this method works, we form pair of characters similar to compression encryption and then we form two characters as cipher output, if there are odd numbers of characters in some layer L then we form number of pairs and the last character forms the pair with the first character in the text. But the key is prepared in such a fashion that when it is decrypted it only shows that character. Again a unique number must be written before beginning of the key as to represent the change in encryption scheme. Unlike compression encryption, there is only one case associated with expansion encryption that is:

1) Non-identical and identical characters forming pairs:

This case involves all the three cases which was discussed in compression encryption, they are Characters in different row and column, characters in same row and different column and Characters in same column and different row, and also identical characters (we consider character to be identical when i = j = k = l satisfy) forming pairs, there is just one rule for encrypting all of these four cases. First we form pair with the next character in the text and then we form two characters using this pair and then we generate key in such a fashion that it decrypts only the desired character. First we form pair with the next character in the text and then we form two characters using this pair and then we generate key in such a fashion that it decrypts only the desired character. For encrypting planitext character P 1 we use $[P\ 1]_{i;j}[P\ 2]_{k;l} = [C1]_{(k;j)}[C2]_{(l;i)}$ where either (i or

k) or (j or l) or all of them can be equal or different.Now coming to the key part since we are only interested in retrieving our original P 1 thus our key is $key_{decryption}$= il where is a unique number which specifies that the expansion encryption is used (this is similar to that of character in same column different row case in compression encryption here we use a number 1 greater than that for instance if we are using 9 to represent change in order of writing of key, then we use 10 for representing that expansion encryption scheme is used).

Example

Let the plain text which is to be expanded be SW AP N IL then we are to expand individual characters, so in order to do that we form pairs as follows SW AP NI here since L is the last character and since it is not forming pair with any other character, thus we form the pair with S again. Thus here the pair becomes LS LS Thus when expanded the first pair becomes ^{0}= and then second third and fourth pairs become 7, T =, D^0 respectively. The decryption key becomes k =10(31672046) where 10 is used to represent that expansion encryption is used. Thus k =
001101101001111000110111000100100001
1011010011010011010110111100011100011010101

5.Compression and expansion encryption

As it was stated earlier that when compression is accompanied with expansion then the system works more efficiently. This scheme follows the rule of compression and expansion encryption. Let us encrypt SWAPNILS using compression and expansion encryption scheme so first we form pairs thus the pairs formed are SW AP NI now compression cannot be performed further as LS is not forming pair with any other character. Thus we perform expansion encryption here, thus LS pair is formed and is expanded. The benefit of this method is that both the schemes are performed simultaneously and if we find a function (to perform on the key and ciphertext) such that it prevents the cryptanalyst to estimate the key and the ciphertext and also that the compression and expansion can be used in such a fashion that net length of the cipher text is equivalent to that of plain text. So inorder to do that we perform confusion and diffusion and hide characters of ciphertext and number of keys a layers under. Let us understand how confusion and diffusion is performed in the next section.

6.Confusion and diffusion

In order to maintain secrecy of the method used and prevent cryptanalysis we use confusion and diffusion method, in this method the actual characters of plain text are hidden a layer under. Confusion and diffusion is performed by initially generating n characters before every cipher text character and generating n character if desired after the last character. Generalized result is as follows:

i=n

X

$C_{Cd} = C_C + C_{Cfi}$ mod(M XM) i=1

where C_{Cd} is the net cipher i=1 i character output after performing confusion and diffusion on the character(it is not a necessity that the function performed is summation it can again be of a complex types, but the function used must be same by both the parties) which is again a function of h(CC, fC)(it is in the starting of the key which specifies number of fake characters and fake numbers which are going to be used in the ciphertext and plaintext, where ciphertext and plaintext will be equal). Again Cfi is the number of fake characters before the actual cipher character. It can be concluded that all the characters which are displayed are fake characters, as none of the text will contribute to a positive analysis. This method will work best when length of ciphertext is less than that of plaintext then we can generate fake characters and make the plaintext and ciphertext equal in length. The same is done with the key fake numbers are generated while modulo which is taken is of M and numbers which are generated lie from 0 to m 1 and then again the similar function h(KN, fN) is performed where KN is the key number and fN is number of fake numbers before actual key number. Another interesting way by which function can be made unique is by specifying a certain operation which is performed right after specifying number of fake characters in the cipher and plaintexts in the key.

Example

Let the actual key be K = 1223455423413234234 (these numbers will be dis- played as there map locations in the actual key, we are writing them as is for simplicity purpose) and since it is desired that 5 fake characters must be produced before every cipher character or key number. Thus, our key becomes K = 51223455423413234234 and if instead of performing summation multiplication is desired then our key becomes K = 51223455423413234234 although the key will consist of address of each number and fake numbers displayed in the key. Let our net key after generating fake numbers become K= 52345112345122345122345134234515234515
2345142345122345132345142345112345132345122
3451323451223451323451423451223451323451423
45

then we will perform our function and net result will be varied and the key now will be sent through an insecure channel. While characters are concerned then we perform function on the location of these characters and the fake characters which were generated.

Note

The fake characters can be created in such a fashion that the displayed cipher appears to be a null or an open-letter cipher.

7.Making arrangements equal

However with cryptanalysis the cipher can be broken down, thus we change the arrangement after fixed number of encryption of characters which makes it tougher for analysis since now there is change of arrangement often(with proper applied techniques one can even shift from MXM(or MXT) arrangement to NXN arrangement, this will make this scheme more secure and reliable). This is how arrangements are made equal. First Alice announces her arrangement publically now the characters which are involved in her arrangement is known publically now a secret keys are generated with the help of Diffie and Hellman key exchange such that it forms a matrix which at max is equivalent to the arrangement, now product of multiplication of the key matrix and Alices private arrangement is sent to Bob, Since bob has access to key matrix he can retrieve back Alices arrangement, and now both the parties can share a secure communication even via an insecure network.

8.Conclusion

We have come up with a method that can perform 3 types of encryption schemes, can shift from one arrangement to other and is reliable and it uses different keys, each key type has a unique function in the system, these were established above.

Acknowledgment

None.

Conflicts of interest

The author has no conflicts of interest to declare.

References

[1] Kahate A. Cryptography and network security. Tata McGraw-Hill Education; 2013.

[2] Lee HY, Wang NJ. The implementation and investigation of securing web applications upon multi-platform for a single sign-on functionality. International Journal of Advanced Computer Research. 2016;6(23):39-46.

[3] Roy A. Brief comparison of RSA and diffie-hellman (public key) algorithm. 2016; 1(1):28-31.

[4] Li N. Research on Diffie-Hellman key exchange protocol. In international conference on computer engineering and technology 2010 (pp. V4-634). IEEE.

This paper is selected from proceedings of National Workshop on Cryptology-NWC 2016 organized at JNN College of Engineering Shimoga, Karnataka, India during 11-13, August 2016.