International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-3, January 2020

New Polyalphabetic Substitution Scheme for Secure Communication

Ranju S. Kartha, Varghese Paul

Abstract*: The internet is a very powerful and useful tool for communication, information and connectivity. So it is very important to keep yourself safe and secure online. The best way of secure information is encryption; there are many cryptographic algorithms available for encryption. These cryptographic algorithms are classified according to their encrypting process; as substitution cipher or transposition cipher. In Polyalphabetic ciphers, the substitution rule changes continuously from character to character according to the keyword and plaintext. Vigenere cipher is considered to be the most efficient Polyalphabetic substitution cipher. But it is vulnerable to attacks, due to its repeating nature of the keyword. To overcome this vulnerability, here we are presenting a new Polyalphabetic substitution scheme which uses infinite number of 26 x 26 random tables for encryption. During encryption, whenever the keyword repeats, this proposed Polyalphabetic substitution cipher generates a 26 x 26 alphabetical random table. Instead of using the same Vigenere Table here we are using an infinite number of alphabetical tables depending on the length of the plaintext and keyword. Each random table will be completely independent from the previous table. This will reduces the repeating sequences in the ciphertext. The repeating nature of the keyword does not help the crackers to break this code. So this proposed Polyalphabetic substitution cipher is considered as an unbreakable cryptosystem. The Proposed Polyalphabetic cipher can provide security for many applications such as web transactions, web transactions, personal emails, secret information transmitted between public or private organization, military application etc.

***Keywords* : Polyalphabetic Cipher, Vigenere Cipher, Vigenere Table, Kasiski Method, Index of Coincidence IC.**

I. INTRODUCTION

Security means protecting something from unwanted or undesired. In modern world all our data which exists in the cyber space is very confidential and needs to be secured in very efficient manner. Internet is liable to be attacked by anyone from anywhere on the planet. Many organizations store data of their costumers online for providing fast services. These developments generate a growing concern for privacy and data security. The organization is responsible for the security of data so that unauthorized persons cannot access and modify the data.

Revised Manuscript Received on January 30, 2020.
***** Correspondence Author
Ranju S. Kartha*, School of Computer Science, Mahatma Gandhi University, Kottayam, India. Email: ranjuskartha@gmail.com
Dr. Varghese Paul, Department of CSE, RSET, Cochin, India. Email: vp.cusat@gmail.com

Data security is more challenging and complex in many of the applications. If an unauthorized person gets this data, the whole organization may suffer irreparable or permanent loss. One of the efficient methods for solving these security issues is Cryptography.

The main term Cryptology is the science and secret communications involving both Cryptography and Cryptanalysis. Cryptography is the technology that used for transmitting messages to make them secure and immune to attack. Cryptanalysis deals with study and analysis of breaking the secrecy of the message with or without knowing the key. Confidentiality, Integrity and Availability are the main goals of cryptography. Cryptographic algorithms are to be designed to achieve the above goals. Cryptographic algorithms rely on a secret piece of information which is known as secret key. The algorithm is known to everyone that exists in public domain but the secret key which is only known to the authorized persons. The objective of any attacker is essentially to find out the key. The attacker is known as cryptanalyst and cryptanalysis is also referred as the art of finding the secret key. People use the cryptographic algorithm for encrypting the secret messages on the hope that the information inside the encrypted message is very secure for sending it to another person. The attacker not having the secret key, tries all possible combinations to break the security of the messages. If he is not successful, he will find some other means to break this cipher. It is possible due to the shortcomings of the cryptographic algorithm or because of the problems occurring while implementing the cryptographic algorithm.

II. POLYALPHABETIC SUBSTITUTION CIPHER

In Polyalphabetic substitution cipher, instead of using a single substitution, it uses several substitutions. Polyalphabetic substitution cipher, as the name suggests more than one substitution rule is applied during the construction of the ciphertext. It has two rules, first one is a set of Monoalphabetic substitution rules and the second one is the key decides which particular Monoalphabetic substitution is to be performed for a given transformation. In this case the mapping between plaintext and ciphertext is one to many.

For example if same letter occurs in first and fifth position in the plaintext it will be encrypted as different letters in the ciphertext according to the key. Here the encryption rule changes continuously from letter to letter according to the key and the plaintext. So it hides the letter frequency of the English language.

A. Vigenere Cipher

The best known Polyalphabetic substitution cipher is Vigenere Cipher. In Vigenere cipher the substitutions are based on a 26 X 26 alphabetic table called as Vigenere Tableau, Vigenere square, or Vigenere Table [1].

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

Fig. 1.Vigenere Table

For constructing this table first write A to Z in alphabetic order in row wise as the row heading on the top of the table. Again write A to Z in alphabetic order in column wise as the column heading on the left side of the table. Then fill the first row of the table by writing A to Z in alphabetic order. The next row is filled by shifting of the first row left by one position in a circular way. The remaining rows of the table can be filled by the circular left shift of the previous row. So each row and column in the Vigenere table having A to Z without any repetitions.

In Vigenere table, the row heading corresponds to the keyword letters and accordingly we have to fix each letter in the row. And the corresponding plaintext letter must be located in the column heading. The intersection of the corresponding row and column will give the ciphertext letter. And repeat this process until the entire message is encrypted [1].

For example consider a message to be encrypted is COMING ON SUNDAY and the keyword is TIME, during encryption the keyword repeats until it is equal to the length of the message as shown in the table below.

Table- I: Encryption Process

Plaintext	C	O	M	I	N	G	O	N	S	U	N	D	A	Y	
Keyword	T	I	M	E	T	I	M	E	T	I	M	E	T	I	
Ciphertext	V	W	Y	M	G	O	A	R	L	C	Z	H	T	G	

For decryption first locate each keyword letter in the row heading and find the position of the corresponding ciphertext letter in that row. Then the letter on the top of the column contains the ciphertext is the plaintext letter. And repeat this process until the entire plaintext is recovered. Here the same letter in the plaintext need not be encrypted as the same ciphertext letter; it will depend on its keyword letter. In Vigenere cipher each message letter can be encrypted as any of 26 letters. If the length of the keyword is m, then the given message can be encrypted in 26^m ways.

B. Cryptanalysis of Vigenere Cipher

In monoalphabetic cipher the cryptanalyst maps the most commonly occurring letter in the ciphertext with the letter E or T. But this is not possible in the case of Vigenere cipher. Here the encryption process hides the relative letter frequency of the English language. So it is very difficult to break this cipher using statistical analysis. As the message can be encrypted in 26^m ways, the brute-force attack is not possible to crack the Vigenere cipher [1].

Mathematical Cryptography began when Friedrich W. Kasiski published a method of breaking Vigenere cipher in 1863 [2]. The fundamental weakness of Vigenere cipher is that if the key length is known the ciphertext can be split apart into individual shift cipher. So the security of the Vigenere cipher lies on having the key length unknown. If we know the key length, by applying statistical analysis, we can break this cryptosystem. This vulnerability happens due to the keyword repetition. Keeping the key length secret is absolutely critical.

Cryptanalysis of Vigenere cipher includes two steps: first step is to find the keyword length and the second is to figure out the letters in the keyword. There are two methods for finding the length of the keyword. The first method is Kasiski method - to find the keyword length using the repeated text sequences in the ciphertext and the second method is Index of Coincidence - to predict the number of alphabets used for substitution.

In 1863, a German military officer Friedrich W. Kasiski published his book describing the Kasiski Test for attacking the Vigenere cipher by finding the length of the keyword used for encryption and decryption process [3]. Kasiski have the following insights: (1) There are many repeated bigrams and trigrams in the plaintext. For example TH, RD, ED are the common bigrams and THE, END, AND are the common trigrams. (2) From time to time, two occurrences of bigrams/ trigrams will be separated by an exact multiple of the key length. This means that the two occurrences will be encrypted in the same way.

Kasiski test based on the above idea, explained that two identical plaintext segments will be encrypted to the same ciphertext whenever they appear d positions apart in the message, then these identical plaintext segments should be encrypted approximately d/m times from the same alphabet, where ($d \equiv 0 \pmod{m}$ and m is the keyword length [3]. So Kasiski suggests for searching the ciphertext for repeated identical segments of length three or more. Then record the distance between their starting positions such as d_1, d_2, $d3$…etc. The keyword length m should divide all d_i s. That means m divides the Greatest Common Divisor (GCD) of these d_i s. So after finding the distances of repeated segments we can determine the GCD of these distances d_1, d_2, $d3$…etc. and the keyword length will be one factor of this GCD.

The second method for finding the keyword length was discovered by William Friedman. The Friedman's Test helps the cryptanalysis of Vigenere Cipher which is based on the value of Index of Coincidence (IC). Index of Coincidence can be used to determine the keyword length m as well as to confirm the keyword length m determined by Kasiski Test.

It can indicate whether the Polyalphabetic substitution cipher is used. The Index of Coincidence is the probability that any two randomly selected letters of the string are same [3]. It is based on probability of randomly selecting two identical letters decreases as the key length increases.

Suppose X = x_1, x_2, x_3...x_N is a string of length *N*, Index of Coincidence of X is denoted by $I_C(X)$ is the probability that two random elements of X are identical. Then the following equation is used for calculating the Index of Coincidence:

$$I_C(X) = \sum_{i=1}^{c} n_i (n_i - 1) / (N(N-1)) \quad (1)$$

where N is the length of the string and n_1 through n_c are the frequency of the letters of the plaintext. And c = 26 for English language.

For a string of English text, the probability that two random elements (x_i) are same is p_i^2. So $I_C(x_i) \approx 0.068$. For shift cipher every alphabet is just permuted, the frequency distribution is not changed. Therefore the IC value of the ciphertext is not changed. In the case of Monoalphabetic substitution cipher the IC value of the ciphertext is around 0.068, because the individual probabilities will be permuted, but the $\sum p_i^2$ will not be changed. So this is an invariant. This property is used to determine whether the cipher is Monoalphabetic substitution or Polyalphabetic substitution. A text of randomly chosen letters will have IC value as low as 0.038. The frequencies of the letters in the ciphertext obtained from Polyalphabetic substitution cipher, are not uniform so the IC value is closer to 0.038.

The Index of Coincidence value can also be used to find the length of the unknown keyword. The cryptanalyst/ attacker can guess the keyword length *m* and divide the ciphertext into *m* substrings and they are referred to as cosets. Then the attacker will begin from *m* = 2, 3, ... and compute the IC value of each substring. The attacker also computes the average IC value corresponding to each value of *m*. This process is repeated until the average IC values of these cosets would still be high and close to IC value of the typical English language 0.068. Otherwise the average of IC's would be low. Based on these observations the attacker can guess the keyword length. The length that yields the highest average Index of Coincidence value or the value closer to 0.068 is likely to be the correct length of the keyword.

After finding the keyword length the attacker can easily decrypt this ciphertext using statistical analysis. If the keyword length is *m*, then every m^{th} character of the plaintext is encrypted using the same shift. If we take the m^{th} character and calculate the frequencies, we should get the frequency in a permuted order. Once the length of the keyword is found, the cryptanalyst writes the ciphertext in columns according to the length of the keyword. Each column containing the ciphertext can be considered as the monoalphabetic cipher, encrypted with the same key. By frequency analysis we can easily decrypt this ciphertext. Using Friedman test or Kasiski method and statistical analysis the cryptanalyst can easily break the Vigenere cipher. This is because of the repeating nature of the key.

People use the cryptographic algorithm for encrypting the secret messages on the hope that the information inside the encrypted message is secure for sending it to another person. The importance of cryptographic algorithms is increasing because of more inclusion of smart devices in our daily life. The cryptographic algorithm plays a crucial role in securing many of the applications. For enhancing security most of the applications uses cryptographic algorithms. All modern cryptographic systems uses symmetric-key encryption algorithms internally to encrypt the bulk of the messages, but they eliminate the need for a physically secured channel by using Diffie–Hellman key exchange or some other public-key protocol to securely come to agreement on a fresh new secret key for each message. So the cryptographic algorithm will continue to play a very crucial role in securing all aspects of our technical world. Hence the problem was identified and the modification of Vigenere cipher is suggested to enhance the security without compromising the security of the traditional Vigenere cipher. The main aim of this research work was to find an efficient way in which the repeating nature of the keyword will not help the attackers to break the Vigenere cipher.

III. PROPOSED CRYPTOSYSTEM

In this proposed cryptosystem the keyword is repeating until it is equal to the length of the plaintext. Instead of using a single Vigenere Table, here we propose to use multiple numbers of 26 X 26 tables. Whenever the keyword repeats during encryption, the proposed Polyalphabetic Substitution Cipher generates a random 26 X 26 table. That means here the multiple number of 26 X 26 tables are used for encrypting the plaintext based on the length of the plaintext and the keyword. The randomly generated 26 X 26 table will be exactly different from the previous tables. These tables are having 26 rows and 26 columns and each row and column will be unique in nature. So here we are using the concept of composition table and addition table in mod 26.

A. Generation of 26 X 26 Random Tables

A 26 x 26 random table can be generated as follows:

- First construct a composition table by generating 26 elements from 0 to 25 in random order and it will be the row heading of the table. Again generate another 26 random elements from 0 to 25 in random order and it will be the column heading of the table.
- Now compute the entries of the table by performing the modulo addition in mod 26. First column heading is added with each of the row heading and divide it with 26, then we will get the second row. Similarly second column heading is added with each of the row heading and divide it with 26, now we will get the third row. This process is repeated until the entire column heading is added with each of the row heading. Finally we will get a 26 X 26 table.
- We can see that, here all the entries of the table are the elements of the set 0 to 25. And any of the row or column has no repetition of elements. Thus all the rows and columns will be unique in nature. And all the rows and column having 0 to 25 in different order.
- Finally convert the entries of the 26 X 26 table from numeric values to alphabetical letters. Now we will get a complete random 26 X 26 alphabetical table for encryption purpose in the proposed cipher. We are not considering row heading and column heading as part of encryption table.

- During encryption and decryption process, modify the row heading and the column heading by including letters from A to Z in its alphabetical order to represent the plain text letters and keyword letters respectively. A randomly generated 26x 26 table is shown in Fig. 2.

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

Fig. 2.26 X 26 Random Alphabetical Table

B. Encryption and Decryption Algorithm

In the case of Vigenere Cipher, the first and the $(m+1)^{th}$ letter of the plaintext are encrypted using the same keyword letter and the Vigenere Table. But in this proposed Polyalphabetic cipher the first and the $(m+1)^{th}$ letter of the plaintext are encrypted using the same keyword letter but encryption table is different. So the mapping between plaintext and ciphertext will be different. This reduces the occurrences of repeated sequences in the resultant ciphertext. The repeated sequences occur in the ciphertext because of mere coincidence.

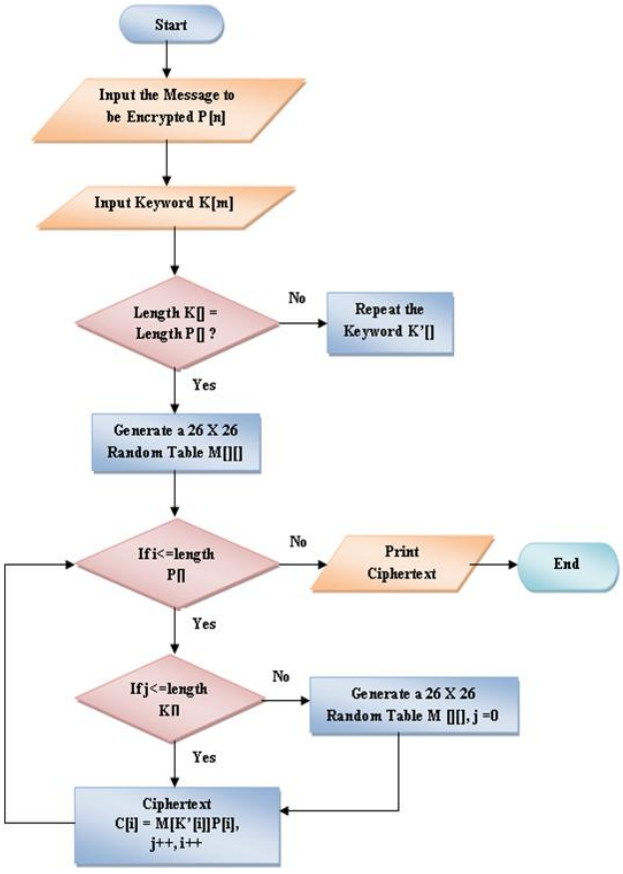

Fig. 3.Encryption Algorithm

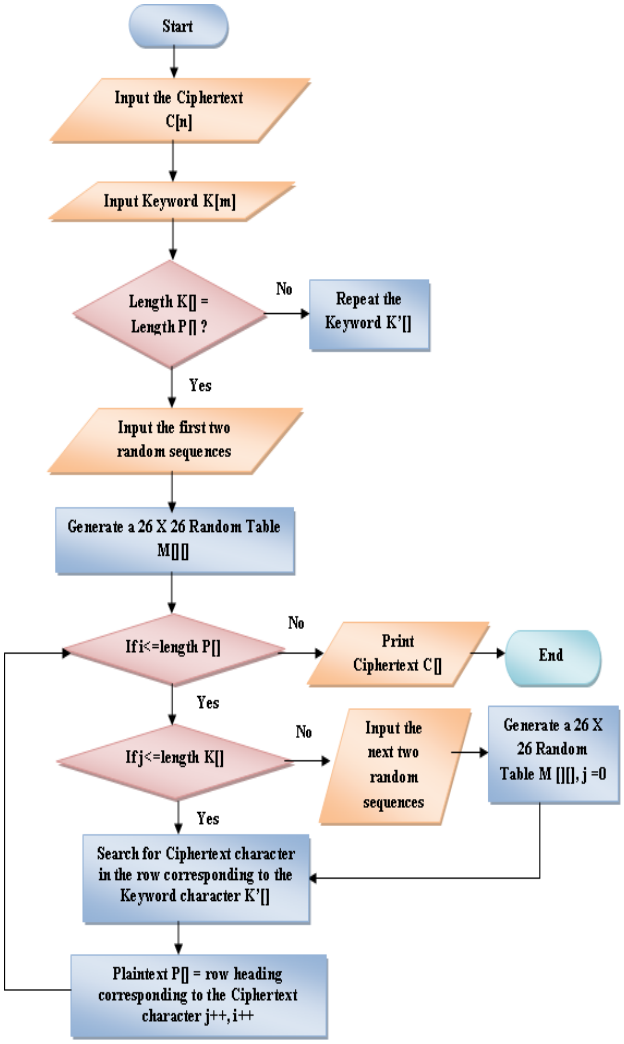

Fig. 4.Decryption Algorithm

During decryption the proposed Polyalphabetic Cipher requires the keyword and the random tables. So the sender should share the keyword along with the sequences required for generating random tables through a secured channel. After decrypting *m* ciphertext letters, the proposed Polyalphabetic substitution cipher again generates a random 26 X 26 table from the next two random sequences received from the sender. Then repeat all the above steps for decrypting the next *m* ciphertext letters. This process is repeated until the end of the ciphertext letters. The flow chart for the proposed encryption and decryption process can shown in Fig. 3 and Fig. 4 respectively

IV. RESULT ANALYSIS

As the technology grows without any boundary, people are using mobiles, computers and internet for most of the applications in their daily life. This explosive growth in technology also helps the attackers to find new loopholes in the security of the applications. So all these applications should have the solution for its security issues. Here we can explain how the proposed Polyalphabetic Substitution Cipher can solve all these security vulnerabilities.

A. Cryptanalysis based on Friedman's Test

By using Friedman Test the attacker can find the length of the keyword. In the case of proposed cipher, it is impossible to find the length of the keyword by using Friedman Test. For explanation, we have chosen a plaintext having large number of repeated sequences.

Consider a plaintext P1**:**

DISCA RDEDC OMPUT ERSDI SKDRI VESAN DMEDI AAREA LSOAP OTENT IALSO URCEO FPLAI NTEXT SMOST OPERA TINGS YSTEM SDONO TACTU ALLYE RASEA NYTHI NGTHE YSIMP LYMAR KTHED ISKSP ACEOC CUPIE DBYAD ELETE DFILE ASAVA ILABL EFORU SEAND REMOV EITSE NTRYF ROMTH EFILE SYSTE MDIRE CTORY THEIN FORMA TIONI NAFIL EDELE TEDIN THISW AYREM AINSF ULLYP RESEN TUNTI LOVER WRITT ENATS OMELA TERTI MEWHE NTHEO PERAT INGSY STEMR EUSES THEDI SKSPA CEWIT HEVEN LOWEN DCOMP UTERS COMMO NLYSO LDWIT HMANY GIGAB YTESO FDISK SPACE ANDRI SINGM ONTHL YTHIS LATER TIMEM AYBEM ONTHS LATER ORNEV EREVE NOVER WRITI NGTHE PORTI ONOFA DISKS URFAC EOCCU PIEDB YADEL ETEDF ILEIS INSUF FICIE NTINM ANYCA SESPE TERGU TMANN OFTHE UNIVE RSITY OFAUC KLAND WROTE ACELE BRATE DPAPE RONTH ERECO VERYO FOVER WRITT ENINF ORMAT IONFR OMMAG NETIC DISKS AREAL STORA GEDEN SITIE SHAVE GOTTE NMUCH HIGHE RSINC ETHEN SOTHI SSORT OFREC OVERY ISLIK ELYTO BEMOR EDIFF ICULT THANI TWASW HENGU TMANN WROTE

And we have chosen the keyword (K1) as CODEMAKER and executed the encryption program for the traditional Vigenere Cipher. The resultant ciphertext C1 is shown in the figure below.

FWVGM RNIUE CPTGT OVJFW VOPRS ZVUOQ HYENM RCFHE XSYEG QHHRF IKPJQ IUGQO PTCCW QXQXD WDQGW SBEBE KKBJW KSDID URRRA TKGKW OOPKE BEJGO QCFHS RXVVH CEIWT CAADV WTRIU KGNWB AMIFE QXTUE NFPCR HPQTO HWKZH EEAFE ZNOEP QFYVL USDRP ROQFX SLXEE XXIAT USYTR IWKZH WKSDI DFWUI OTYVP VVHMZ FYVDC HLSZI XEWKZ HHQLO XVFWQ XTICA RAFHQ MIXWW WZOCB ROWVP HXRFI VSMGF ZVUTD IECHV SYEVE KGFWM YEGLV PHKIA POVRV WQKEY CXVOF HYEEC XYGRL WWSZE TGKLX TEFIE NCZIZ DMSDR IWIDS MSDOC QPKSY PUYWW LYAXC XKUDF KTOWF HRLWW SZETG OQHDI CMEIA RRFHV CKJWV PMTOV KKAHQ MYLID QBWLE LKXVT CURQV OVVXS QSHEB AIKHL RSTRI GQFWM ANYJR FWVOE UBJRE SRGOU ZMVFP BEPEV IKGRI MXESW ZPGXJ RIMMV PHLRY AXCTC GHWBE DIIII WQMNX SWVVH YZIFI IUWWC AFKYT MZDRP WBSKG OFIXE LVRVS GTMPO VFPHK IDEMS MGFBS ROFII YFLXF EXMEH CUQMT SSEHF RQYAQ RVVWF HUSUW RTSDP ETYVR ISGIZ SSXZG GKEHE QSKVS QQGCR LZIVH VEIXG VVVHR EODLZ UGRVF OPVVE CYIDY SWCKY HPKTY FVOCU IPIPJ ZEIOX FHKRZ VKDWI HORXW HPEZN GVFVS

The above ciphertext is analyzed by calculating the Index of Coincidence (IC). The attacker will guess different values for *m* for conducting Friedman's Test. Here we have given *m* = 1 to 10 and divide the ciphertext into *m* cosets. On the given value of *m*, calculate the IC value on each coset. We found that for *m* = 9 the average IC value is 0.06571, which is approximately equal to the IC value of English (0.68). For all the remaining cases, the IC value is around 0.04 which is approximately equal to the IC value of the random text (0.038). So we can choose *m*= 9 as the keyword length. So in this example we can easily find the correct keyword length using the Friedman's Test.

Next the same plaintext P1 and key K1 are input to the encryption algorithm of the proposed Polyalphabetic substitution scheme and then it's the resultant ciphertext C1' is shown below.

VCPZG VKVTD GUOMM RNOEM QHNVE SIDLC DFDKL CJAYK TOYEY HFDKE PYQPD WIUWQ RLFRW VBANY BGJBY NCCED GDXRX UTFUH ZGVKV MJKUU TZNUP AKBSF JAZBX GWSNO JWQPJ MRIHW BZETJ QBNZS BSZWE MWRUF URIMW LVNGE IBONN SGTQK OMWBF JYFVM HSQVF NCVUR NYDAV PYPJK HSDCE IIQXP BQTKL PZAIF JEXBX PEBXJ LRENJ EOXCC BIXUW LVWDK LFRHI MKXPN XULOD IGJIK ARAEH QYSWA PQLAM UDMIN IJRQD EXRKO JRWZV FIHYF WDGKC PEWOS USVLR WTARU XGXVM PHIYD ZXSMH XROOD DKJPY DNHYJ CUFYN LHYJH JEXWI NBRCP FJJGX PTUDL EWADF FAVMP HIIDZ IOENN HUEDG NGGEC QPEIP NLKAD GENNW MKSJD JHHFJ MESXV PJOGV ODGSU ZEISR VHNTI MGTAO OAHQF OFZNR KTPNK MMVCK BZCCZ MIEVT FXNTR QOCIV ZOIRT FLFQR BYHLU VFIMG JTLPZ MQWPA RMONG UFMKZ LYYCK WTDHA ONOBF ZFUSJ ZJMLG RKVYE XOAYI RJCKG ZSDKW DGFFI HYURH YSXRX IYBFO NQRMH ERLPD AJOMK ATYDA RGMJG BWVBP AULIP FKPZL LMUTG KGZSY DITXL XQHCS RBYET ZEMQV OZLGK WQMHL SVGKW GMBMZ LOQTQ EBBGV BJTYN STFIN LWLWA CUVLJ OYRHG RLBUJ HECWL CSLPO CDWTS IKNDZ WGFCZ CUTBS PZANW

The above ciphertext is analyzed by calculating the Index of Coincidence (IC) as shown in Table 3 below. Here also we are guessing different values for *m* for conducting Friedman's Test. The ciphertext is divided into *m* cosets for each value of *m*. The IC value is calculated for each coset and computed the average IC value corresponding to each keyword length; it is shown in the Table 3. From the table we can see that for all values of *m* the average IC value is around 0.037, which is approximately equal to the IC value of random text (0.038). So this confuses the attacker to guess the keyword length. In the case of Vigenere Cipher we can easy guess the keyword length using this test. But here we can conclude that it is impossible to find the correct keyword using the Friedman's test.

Table- III: Average IC value corresponding to m = 1 to 10 – Proposed Polyalphabetic Cipher

Length m	IC value of the cosets	Average IC
1	0.03774	0.03774
2	0.03754, 0.03802	0.03778
3	0.03712, 0.03834, 0.03658	0.03735
4	0.03719, 0.03883, 0.03757, 0.03836	0.03799
5	0.03781, 0.03854, 0.03528, 0.03609, 0.03830	0.03720
6	0.03829, 0.03887, 0.03664, 0.03758, 0.03699, 0.03876	0.03786
7	0.03524, 0.03716, 0.04247, 0.03555, 0.03571, 0.03829, 0.03620	0.03723
8	0.03772, 0.03997, 0.04103, 0.04082, 0.03850, 0.03850, 0.03598, 0.03745	0.03875
9	0.03683, 0.03918, 0.03929, 0.03635, 0.04063, 0.03716, 0.03234, 0.04117, 0.03796	0.03788
10	0.03635, 0.04025, 0.03635, 0.03505, 0.03830, 0.03596, 0.03763, 0.03430, 0.03230, 0.03963	0.03661

B. Cryptanalysis based on Kasiski Test

Next we analyze the resultant of the encryption process using Kasiski Test. The program was done in MATLAB and the analysis of the result is shown below.

Here we have used the same plaintext P1 and the key K1 for the encryption program of the traditional Vigenere Cipher and found the repeated sequences in the resultant ciphertext C1. The repeated sequences of length three or more are shown in red colour the ciphertext C1. First we have noted the position of the repeated trigrams and the distance between the successive repeated trigrams. According to the Kasiski Test, next we will have to find the factors of the distances between successive repeated sequences and the most occurring factor will be the keyword length. By analyzing these distances the most repeating factors are 2, 3 and 9. The keyword of length 2 and 3 are not commonly used in the encryption process.

If the length of the keyword is very small, then it is easily vulnerable to attacks. So here we can choose m = 9 as the keyword length and it is the correct choice.

Next we have done the Kasiski Test on the above ciphertext C1' obtained from proposed Polyalphabetic substitution cipher. Initially we have found all the repeated sequences in the resultant ciphertext C1' and marked them in red colour. The analysis on the repeated sequences of length more than three are done on the next table. The lengths of the repeated sequences are also noted in the following Table IV.

Table-IV: Analysis of repeated sequences of length more than three in the ciphertext – Proposed Polyalphabetc Cipher

Length	*Position*	*Ciphertex*	*Plaintext*
5	3	ZGVKT	CARDE
	90		SDONO
	333	VMPHI	GABYT
	397		MREUS
4	305	FIHY	TERT
	603		TEDP
	593	KGZS	TEAC
	665		AREA

The analysis on the repeated trigrams in the above ciphertext C1' is done on the following Table. The positions of the repeated trigrams are also noted in the Table V.

Table-V:Average IC value corresponding to m = 1 to 10 – Proposed Polyalphabetic Cipher

Positions	Ciphertext	Plaintext	Positions	Ciphertext	Plaintext
18	EMQ	DIS	154	WLV	DEL
686		HAV	244		LED
31	FDK	MED	191	YDA	ITS
26		TEN	637		INF
32	DKL	EDI	214	LPZ	EMD
248		LET	537		CIE
61	LFR	PLA	215	PZA	MDI
250		TED	780		WRO
64	WVB	AIN	220	JEX	CTO
646		ONF	370		UTE
78	EDG	RAT	304	VFI	ATE
412		ACE	530		INS
82	XRX	NGS	310	WDG	MEW
612		NTH	599		EBR
357	HYJ	VEN	469	IMG	ENO
366		COM	532		SUF
407	ENN	ISK	524	RBY	FIL
431		THI	680		SIT
693	GKW	TEN			
702		GHE			

According to the Kasiski Test, next we will have to find the factors of the distances between successive repeated sequences and the most occurring factor will be the keyword length. Here the most repeating factors are 2, 3 and 6. The keyword of length 2 and 3 are not commonly used in the encryption process. Hence we can choose m = 6 as the keyword length and it are found to be a wrong decision. So in this proposed cipher, it is impossible to find the keyword length by using Kasiski Test.

The plaintext corresponding to the repeated trigrams are also presented in the above Table IV. Here we can see that the same ciphertext sequences are produced from different plaintext sequences. This is because of mere coincidence. The distance between the successive repeated sequences occurs due to mere coincidence will not help attackers to find the keyword length. So this analysis will lead the attacker to a wrong decision.

C. Cryptanalysis Based on Statistical Analysis

We have already discussed that it is difficult to break Polyalphabetic substitution cipher using statistical analysis. Even though the distribution is seemingly flat in the case of traditional Vigenere Cipher, its distribution still leaks information.Here we considered the previous example and the resultant ciphertext C1 of Vigenere Cipher are applied to MATLAB program for frequency analysis. The frequency analysis graph is shown in Fig. 5. From the graph we can see that N, J, A and L are the least occurring letters and E, I, V, W and R are the most occurring letters in this ciphertext. Here the frequency of occurrence of alphabetic letters varies from 9 to 54. Here it is difficult to break this cipher using this frequency analysis but it is not impossible to break this cipher.

Next we can consider the ciphertext of proposed Polyalphabetic substitution cipher C1' from the above example and input to the MATLAB program for frequency analysis. The resultant frequency analysis graph is shown in Fig. 6. Here we can see that the resultant graph is almost flat. Here the letter Q is having the least frequency of 23 and the letter N having the highest frequency of 35. So frequency of occurrence of remaining letters lies between 23 and 35. The number of occurrence is common for most of the letters. In this proposed Polyalphabetic substitution cipher, it is impossible for an attacker to compare the ciphertext letters with alphabets according to the frequency analysis. Hence we can say that cipher enhances the security and it is impossible to break this cipher using any statistical analysis.

In the case of Vigenere Cipher the percentage of occurrence of plaintext letters varying between 6.88% and 1.15%. In this case the frequency distribution is almost flat, there is some deviation, and this is because of the repeating characters in the ciphertext. And the case of proposed Polyalphabetic Substitution cipher, the percentage of occurrence of each alphabets lie between 2.93% and 4.46%.

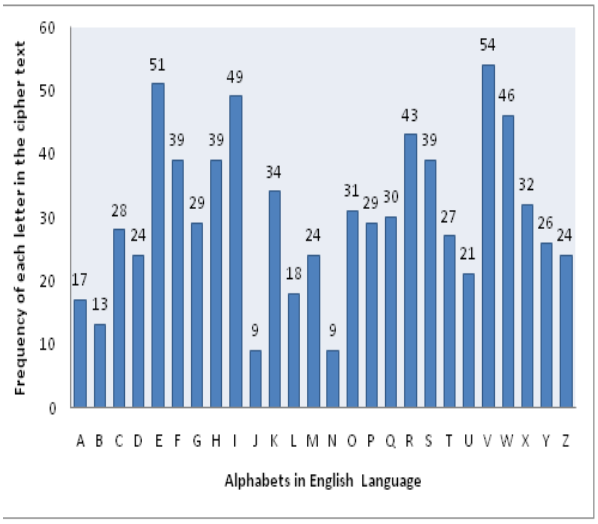

Fig. 5. Frequency Analysis of Vigenere Cipher

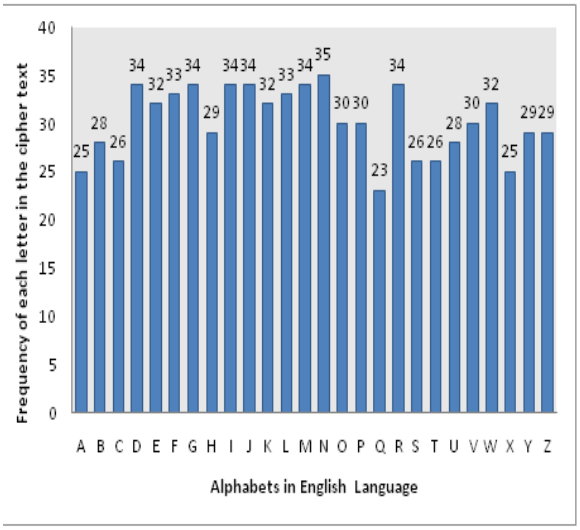

Fig. 6. Frequency Analysis of Proposed Polyalphabetic cipher

The percentage of occurrence of each alphabet is almost same. Here the deviation in frequency analysis is very less so the cryptanalysis of the proposed cipher is impossible. In this case the percentage of occurrence of alphabets in English language and the proposed cipher is entirely different, so mapping will not be possible between the alphabets and letters in the ciphertext of proposed Polyalphabetic substitution cipher. So we can conclude that the proposed cipher is an unbreakable Polyalphabetic substitution cipher.

V. CONCLUSION

The proposed cryptographic algorithm is a new Polyalphabetic Substitution Scheme which overcomes the primary weakness of the Polyalphabetic cipher by using multiple versions of Vigenere Table. The cryptanalysis of the Polyalphabetic substitution cipher was possible if the attacker knows the keyword length. Here we tried two methods for finding the keyword length in this proposed Polyalphabetic cipher. But it is proved that all those methods failed. Thus the security of this proposed cipher does not depend on the length of the keyword, which is used for encryption.

So here we can conclude that this proposed Polyalphabetic cipher is an unbreakable cipher.

The proposed system encrypts only the alphabets in the plaintext. So it can be modified by including all alphabets (A to Z), small letters (a to z), numbers (0 to 9) and symbols (32) present in the keyboard into the plaintext domain [6]. Then the cipher can encrypt all the keys in the QWERTY keyboard. We can also include blank space to plaintext domain. In the proposed cipher we can easily consider the blank space between plaintext letters without any security issues. So we can increases the size of the key domain and the plaintext domain. For encrypting all these we have to generate a 94X 94 random table. The importance of cryptographic algorithms is increasing because of more inclusion of smart devices in our daily life. In today's world most popular electronic equipment is mobile. The communication between mobile and base station should be encrypted. The cryptographic algorithm plays a crucial role in securing many of the applications in mobile phone such as HTTPs, Whatsapp and Digital signatures etc. As cryptography grows without any boundaries and this in turn cause an increase in activities of the cryptanalyst to find new loopholes. Hence cryptology offers an immense potential for research activities.

ACKNOWLEDGMENT

We like to thank the Director of School of Computer Science, Mahatma Gandhi University, Priyadarsini Hills, Kottayam for providing all the facilities to complete the task.

REFERENCES

1. William Stallings. (2005), "Cryptography and Network Security Principles and Practices", Fourth Edition, Prentice Hall .
2. Debasis Das, U. A. Lanjewar, S. J. Sharma, "The Art of Cryptology: From Ancient Number System to Strange Number System" International Journal of application and Innovation in Engineering & Management 2 (4), 2013.
3. Tobias Schrodel, " Breaking Short Vigenère Cipher" Journal of Cryptologia 32 (4), 2008.
4. Ranju S Kartha, Dr. Varghese Paul, " Survey: Recent Modifications in Vigenere cipher", IOSR Journal of Computer Engineering (IOSR-JCE) 16 (2), 2014.
5. Robbi Rahim ,Nuning Kurniasih , M Mustamam, *Liesna Andriany*, Ansari Saleh Ahmar. "Combination Vigenere Cipher and One Time Pad for Data Security" International Journal of Engineering & Technology, 2018.
6. Prof.Ravindra Babu Kallam, Dr. S.Udaya Kumar, Dr. A.Vinaya babu, V.Shravan kumar, "A Contemporary Polyalphabetic Cipher using Comprehensive Vigenere Table" World of Computer Science and Information Technology Journal, 2011.

AUTHORS PROFILE

Ms. Ranju S. Kartha, pursed B.Tech from M. G. University Kottayam in 2006 and M.Tech from Amrita University, Coimbatore in year 2008. She is currently pursuing Ph.D. and currently working as Assistant Professor in Department of Electronic and Communication Engineering, Sree Narayana Gurukulam College of Engineering, Kadayiruppu since 2008. She is a a life member of Indian Society for Technichal Education (ISTE). She has published many research papers in reputed international journals. Her main research work focus on Cryptography Algorithms. She has 11 years of teaching experience.

Dr. Varghese Paul, pursed B.Sc (Engg) from Kerala University and M.Techfrom Cochin University of Science and Technology. He pursed Ph.D in Computer Science from Cochin University of Science and Technology and currently working as Professor in Department of IT, RSET, Cochin. He is a Research Supervisor of Cochin University of Science and Technology, M G University Kottayam, Anna Technical University Chennai, Bharathiar University Coimbatore, Bharathidasan University Trichy and Karpagam University Coimbatore. Under the guidance, 29 research scholars had already completed research studies and degree awarded. His research areas are Data Security using Cryptography, Data Compression, Data Mining, Image Processing and E_Governance. Developed TDMRC Coding System for character representation in computer systems and encryption system using this unique coding system. He has published many research papers in international as well as national journals and a text book also. He has 10 years of teaching experience and 19 years of Industrial Experience.