Modified Vigenere Cryptosystem Using Matrix Manipulation and Base94 Encoding Scheme

Jason P. Sermeno[1], Kenrick Agustin S. Secugal[2], Nelly E. Mistio[3]*

Abstract: Security in the field of Information Technology is the defense of digital information along with its assets against internal and/or external malicious and accidental threats. To secure such valuable information, cryptography is used. A cryptography or encryption algorithm is the process of transforming sensitive data into confounding data in such a way that the person or the machine with a key can decode the hidden information. Among these algorithms is the Vigenere matrix. In this article, a hybrid approach is used in the encryption algorithm. This approach implements the matrix manipulation principle and Base94 encoding algorithm. In order to measure the performance of the proposed algorithm, the Avalanche Effect is used. The result of the study shows a significant high avalanche effect compared to the traditional Vigenere cryptosystem.

Keywords: Vigenere Cryptosystem; Encryption Algorithm; Table Manipulation; Symmetric-Key Algorithm; Security Cryptography

1. Introduction

Cryptography is the study and practice of concealing information through encoding and transformation of data. It is an essential tool to help protect privacy, ensure trust, control access, and perhaps secure electronic transactions as well as managing digital rights. In practice, a plain message undergoes enciphering or series of enciphering and is restored through a decryption process.

There are encryption schemes that are used that constitute the area of study of cryptography. One popular encryption system is the Vigenere Cryptosystem developed by Blaise De Vigenere in 1593. It is a poly-alphabetic cipher that uses tabula recta or a Vigenere table and a custom key to manage the encryption and decryption of the message. So far, it has been best-known poly-alphabetic substitution

[1] College of Computer Studies, University of Antique, Sibalom, Antique, Philippines
Email: jasonpsermeno@gmail.com
[2] College of Computer Studies, University of Antique, Sibalom, Antique, Philippines
Email: kenrick.agustin.secugal@antiquespride.edu.ph
[3]* College of Computer Studies, University of Antique, Sibalom, Antique, Philippines
Email: nmistio@yahoo.com (Corresponding Author)

Acknowledgement: This paper has been supported and funded by the University of Antique.

Received [August 12, 2020]; Revised [October 8, 2020]; Accepted [October 20, 2020]

Published by InnoCon Publishing
ISSN 2704-4440

cipher which is more potent than Ceasar cipher and much harder to decode [1]. But in this scheme, the security of the algorithm relies on the security of the keys used. While it is true that one issue with the keys is the key exchange problem wherein how we could securely communicate a shared key before any secure communication can be initiated. In some situations, direct key exchange is possible, however, much commercial data exchange now takes place between parties that have never previously communicated with one another, and there is no opportunity to exchange keys in advance [2]. Another particular problem lies in the tabula recta or the Vigenere Square itself. If ever the encryption key had been known, the concealed message could be easily decoded using the fixed table that had been constructed.

This study proposes a new approach to enhancing the Vigenere Cryptosystem. It uses the matrix manipulation principle for simultaneous manipulation of the Vigenere table and involves a series of iterative substitution transformations using 2 keys for encrypting and decrypting the message as depicted in Fig. 1. The approach also implements the Base94 encoding scheme that alters the message to a different form so that it would offer a higher means of securing the information. With this new strategy, it should attain a good avalanche effect compared to the traditional Vigenere encryption algorithm.

2. Related Works

The Vigenere Cryptosystem is a popular polyalphabetic substitution cryptographic cipher. It is considered classical today because it is not used now but it had great importance in its history. This algorithm usually operates on letters and is implemented with a simple mechanical device or by hand. They were considered reliable at the moment it was developed but have little importance now due to the technological advancements.

Nowadays, the Vigenere encryption algorithm had been used as one of the mechanisms for encrypting information in their applications for some studies. One particular research is the SMS (Short Message Service) encryption using the combination of the Vigenere and Ceasar cipher encryption scheme for Android phones [3]. The research was just an improvement of a previous study wherein the number of characters was just increased to 94 to give the application a higher number of possible combinations. A similar study had also been conducted using the Least Significant Bit Steganography combined with the Vigenere Cipher for Android platforms [4]. Their approach uses arithmetic coding and a hash function (SHA 256). The hybrid algorithm was able to enhance the security of the confidential data delivery process, reduce the file size, and detect the authenticity of the files being sent. In addition to this, it was able to encrypt different forms of image file (*.gif, *.jpeg, *.png, *.bmp) as well as document type files (*.doc, *.xls, *.pdf, *.txt). Another similar study focuses on digital image encryption using just the Vigenere itself [5]. In their research, the image file along with its key is encrypted using a Red-Green-Blue (RGB) matrix as the Vigenere table. The same principle is being applied for the encryption scheme, only that the table is based on a 255 gray-scale of red, green, and blue.

Aside from having the Vigenere algorithm as one of the mechanisms for a certain application being developed, there are also studies that involve the enhancement of the Vigenere Cryptosystem by merging it with other existing encryption schemes making it more reliable and difficult to crack. One such enhancement involves a scientific procedure on the shifting and substitution transformation while preserving the philosophy of the scheme. It takes advantage of the principle of a discrete logarithm problem on the chosen prime number making it difficult to decode. The strength of the proposed algorithm relies on the prime factor and its relevant primitive root [6]. Another similar study is the fusion of the Advanced Encryption Standard (AES) and Base64 algorithm with the Vigenere Cryptosystem. The hybrid algorithm works its way out on a series of substitution transformations and bit operations using the cipher key. Based on their study, while increasing the number of characters in their table, the arrangements were not according to its corresponding American Standard Code for Information Interchange (ASCII) code. They applied the aspect of the substitution transformation over the message, used the Base64 algorithm to convert it to a different string, and finally used the AES to encrypt the key. The result of their encryption process showed a high avalanche effect on the resulting encrypted message [7].

At the present time, cryptography relies greatly on the theory of mathematics and computer science. By using computational resistance in the designs of such algorithms, it only means that it is impossible to crack it in any current known way [8]. This indicates that faster computing technology and theoretical advancement entails constant modifications to these techniques. However, theoretically secure schemes had been proven to be impossible to crack if used correctly. This means that even with unlimited computing power these schemes still stand but they are very difficult to implement.

3. The Modified Vigenere Cryptosystem

3.1 Generating and Decoding the Tabula Key

The Modified Vigenere Cryptosystem holds four main steps during the encryption and decryption process: 1. generation and decoding of Tabula key; 2. manipulation of the Vigenere square; 3. iterative Vigenere encryption procedure; 4a. message-and-key fusion (during encryption); and 4b. message parsing (during decryption).

A tabula key is a 23-digit hexadecimal number that is randomly generated by the system. Its primary purpose is to provide a pair of swap patterns on the collective rows and columns of the Vigenere table based on the bit patterns of the tabula key in its binary form. A single hexadecimal digit is comprised of four binary digits, so having 23 digits would give us 92 bits. Since this study makes use of a 95x95 square matrix, three more bits will be appended at the end of the binary-string. The way the extra bits are added will be the replica of the last binary digit in the existing set of bits, instead of adding series of zeroes at the end. Table 1 depicts some examples of how a portion of the tabula key is decoded.

Table 1. Decoding the tabula key

Tabula Key	Binary Representation	Extra Bits Padded
B2 … F3	1011 0010 … 1111 0011	1011 0010 … 1111 0011 <u>011</u>
3A … 2A	0011 1010 … 0010 1010	0011 1010 … 0010 1010 <u>010</u>
7F … B1	0111 1111 … 1011 0001	0111 1111 … 1011 0001 <u>001</u>

The row-swap and column-swap patterns are determined by reading the bits from left-to-right and right-to-left respectively where each bit of the tabula key corresponds to the position of the row or column to be swapped.

3.2 Manipulating the Vigenere Square

The normal Vigenere square is based on twenty-six capital letters, followed by a cycle changing the order in a 26x26 square matrix. In order to improve the complexity of the proposed algorithm, while improving the performance on confidentiality, in this paper, 95 characters are taken into account that includes special characters, numeric characters, upper- and lowercase alphabets whose ASCII character codes ranging from 32 to 126 as shown in Table 2. So, it will be difficult for the cryptanalysis to break the algorithm because 95 characters are taken instead of 26.

The idea behind this scheme is that before an encryption or decryption process takes place, the existing Vigenere square is being shuffled using the binary form of the tabula key. Reading the binary form from left to right, we swap the corresponding pair of rows where the occurrences of the pair of nonzero bits are positioned. Reading the binary form from right to left would allow us to swap the corresponding pair of columns in the table.

For example, if the tabula key is C9, its binary form is 11001001. From left to right, the 1st and 2nd rows are swapped, and then the 5th and 8th rows are swapped. From right to left, the 1st and 4th columns are swapped and the 7th and 8th columns are swapped too. If a certain row or column doesn't have a pair, that operation is skipped and proceeds with the next operation.

Table 2. The character set of the modified Vigenere square

Character Description	Symbols	ASCII code range
white space		32
special characters	! ” # $ % & ’ (*) + , - . /	33-47
special characters	: ; < = > ? @	58-64
special characters	[\ ] ^ _ `	91-96
special characters	{ | } ~	123-126
numeric characters	0 1 2 3 4 5 6 7 8 9	48-57
uppercase characters	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	65-90
lowercase characters	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	97-122

The result of this operation should give us a good and nice scrambled Vigenere table that doesn't exhibit the order of the cycle in its original state. An attacker who knows the cipher key but doesn't have a clue about the tabula key would definitely have difficulty in cracking the code.

In order to determine the number of possible pair-swapping combinations that could be done in a square table, we use the formula:

$$\sum_{2\le k\le n,\ \ even} \frac{n!}{(k!\,(n-k)!)} \tag{1}$$

An 8x8 matrix would give us 127 possible pair-swapping combinations. As for the table used in this study, 95 is a big number that would yield a huge number of possibilities. That is, 1.9807x10^{28} possible pair swapping activities. This is a very large number in which it is difficult for the cryptanalysis to figure out the matrix orientation of the table.

3.3 The Encryption Scheme

In this study, the encryption and decryption scheme will be in an iterative pattern encrypting or decrypting both the message and the cipher key using the newly scrambled Vigenere square. Figure 1 shows the conceptual approach of the encryption and decryption process of the proposed algorithm design.

Notice that for every iterative round, two substitution transformations are carried out. This is because the algorithm ciphers and/or deciphers not only the message but the cipher key as well. So, if there are

10 iterative rounds of encryption, the algorithm carries out 20 substitution transformation activities. At the end of the encryption process, the encrypted message, encrypted key, and the table key are fused together as one.

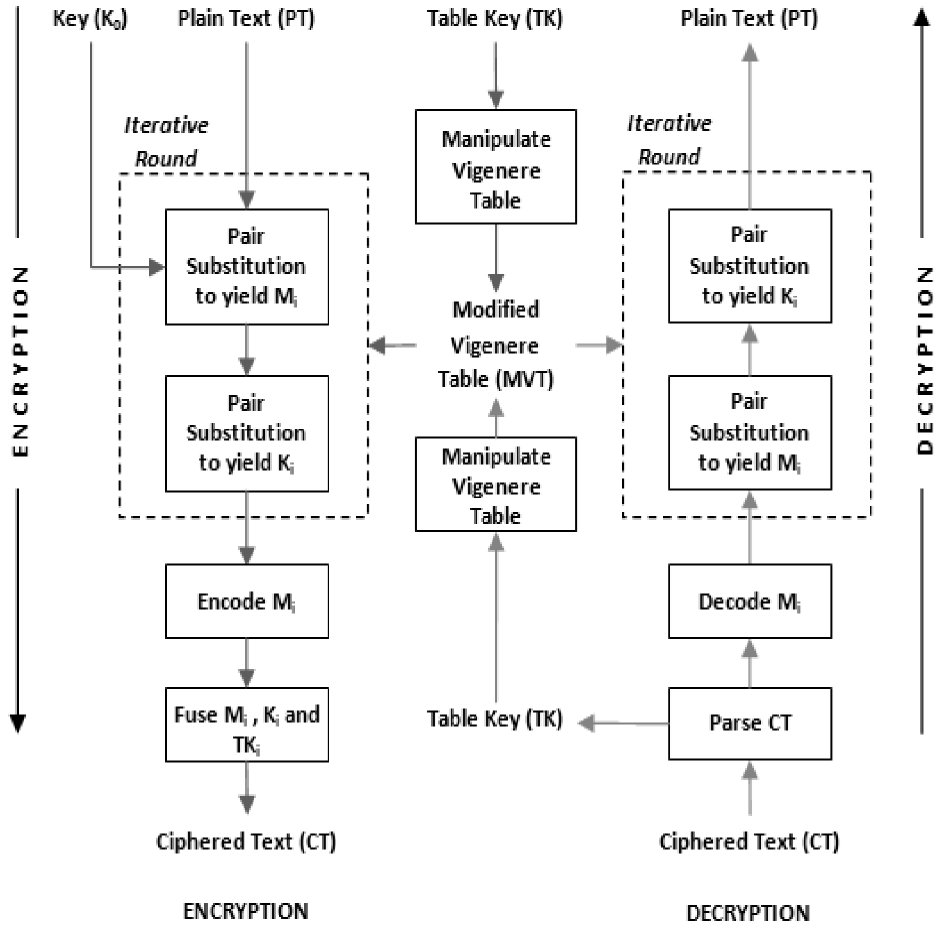
Key (K_0)
Plain Text (PT)
Table Key (TK)
Plain Text (PT)
Iterative Round
Manipulate Vigenere Table
Iterative Round
Pair Substitution to yield M_i
Modified Vigenere Table (MVT)
Pair Substitution to yield K_i
ENCRYPTION
DECRYPTION
Pair Substitution to yield K_i
Manipulate Vigenere Table
Pair Substitution to yield M_i
Encode M_i
Decode M_i
Fuse M_i , K_i and TK_i
Table Key (TK)
Parse CT
Ciphered Text (CT)
Ciphered Text (CT)
ENCRYPTION
DECRYPTION

Figure 1. Conceptual Approach of the Modified Vigenere Cryptosystem

The Modified Vigenere Encryption Algorithm is governed by the following transformations:

1) Binary Conversion of the Tabula Key: As discussed earlier, the tabula key is used for determining the row- and column-swap pattern for the Vigenere table. Here, the tabula key is decoded into its binary form and stored temporarily as a string of 1s and 0s.

2) Pair Swapping Transformation: Upon initialization, the corresponding arrangement of values in the Vigenere table is altered by a series of rows and column swapping activities.

3) Key Substitution Transformation: This transformation is similar to the message substitution transformation. Although instead of working with the message, it encrypts or decrypts the cipher key Ki.

4) Message-Key Interchange Transformation: For both the encryption and decryption process, this procedure allows both the key and message to swap values between the substitution transformations. It consists of 3 simple assignment operations to swap both values.

5) Message Substitution Transformation: This transformation round is a pair substitution of the message and the key to encrypt or decrypt a message. The outcome of this operation is an encrypted or decrypted message Mi.

6) Fusion Transformation: This transformation only occurs in the encryption process. At the end of the iterative encryption process, the encrypted message, encrypted key, and the tabular key are fused together to form one encrypted block of information.

At this point, both keys are shared across the communication channel. An attempt on altering a single character in the ciphered text would definitely affect the plain message, cipher key, and/or the arrangement of the Vigenere table. The output of this transformation process is more than twice the size of the plain text. This is because the cipher key was expanded to the size of the plain text. In addition to this, the length of the tabular key was appended on both ends of the concatenated string.

7) Message Parsing Transformation: The message parsing transformation only occurs in the decryption process. It simply extracts the halves of the tabular key, encrypted message, and the encrypted key. It is the reverse process of the fusion transformation.

8) Encoding Transformation: This method allows the encrypted message to be encoded using the Base94 encoding scheme. The perspective view of the Base94 scheme is quite similar to Base85 algorithm. In fact, Base85 is the root of the design of Base94 algorithm. The only difference is that the scheme uses 94 printable characters in the ASCII character set.

9) Decoding Transformation: This particular transformation occurs in the decryption process and it is used to restore the encoded message to its encrypted form before the decryption takes place. Decoding is done inversely. But when a certain 4-tuple lacks 1 to 3 bytes of data, it will be padded with '~' characters.

3.4 The Algorithm Design

Based on the previous concept and detailed architecture of the Modified Vigenere Cryptosystem, the design of the algorithm is unique compared to other existing encryption algorithms. The keys used in this design are auto-generated and input-based. The Vigenere table itself is not always fixed, it is altered every time the encryption occurs, so the probability of knowing the orientation of the table is pretty high. To understand how the proposed system works, the proponent had come up with a system design for the encryption and decryption process.

As shown in Figure 2, the encryption process starts with a random generation of the tabula key. The contents of the Vigenere table is then scrambled using the tabula key before acquiring input from the user for the plain text and the cipher key as well. Reading the characters from the file requires the size of the cipher key to be equal to the size of the plain text through a replication process. From here, each character of the message will be encrypted using this formula: Ci = (Pi + Ki) mod 95, where Ci is the ith character of the ciphered message, Pi is the i^{th} character of the plain text (or ciphered message on the next iterative round), and Ki is the i^{th} character of the cipher key (or ciphered key on the next iterative round). The encryption process here is a loop operation that performs a substitution transformation on all characters of the plain message. Next, the plain text and the cipher key are interchanged using the following formulas: T=C ; C=KJ-1 ; and K=C, where T is temporary storage for the ciphered text C (T will be used to restore our ciphered text C on the later part.), KJ-1 is the previous ciphered key on j^{th}-1 encryption round and K is the current cipher key in the active iteration encryption round. The next process encrypts all the characters from the cipher key using the formula: Ki = (Ci + Ki) mod 95. After two consecutive substitution transformations are done, the ciphered message is reverted back to its original content through the formula: C=T. The whole procedure from reading the characters from files is repeated for 10 rounds. Once the 10th round is reached, the ciphered message is encoded using the Base94 encoder and is fused with the ciphered key and the tabula key using the formula: TK1+EM+EK+TK2. The outcome of this encryption process is one encrypted file.

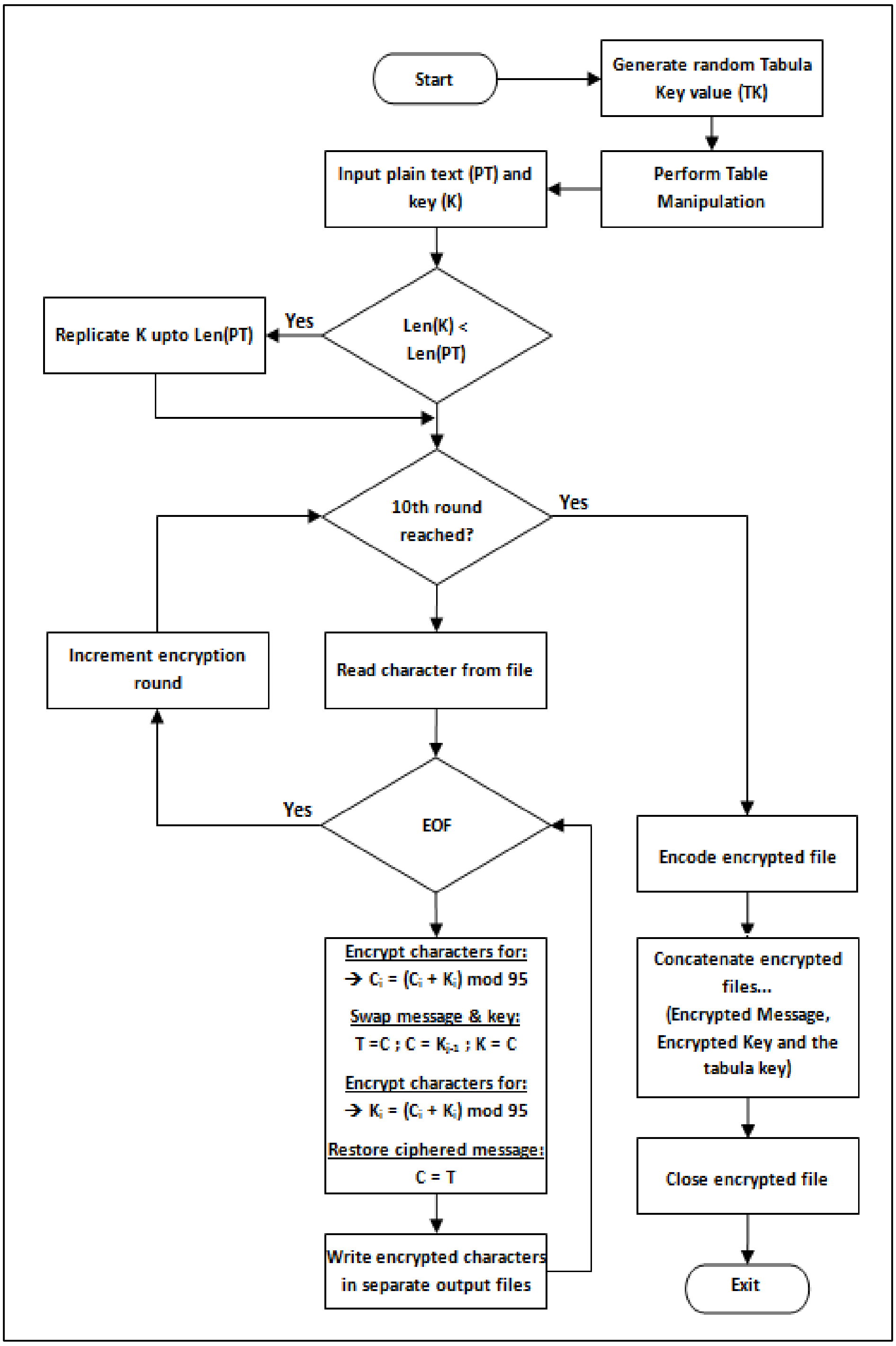

Figure 2. Flowchart of the Encryption Process

However, for the decryption process, as depicted in Figure 3, the procedure is the reverse of the encryption method. The process structure is quite similar to that of the encryption scheme, only that the formulas used are different. It starts with the extraction of the ciphered message and keys of the encrypted file. The ciphered text is then decoded using the Base94 decoder. The contents of the Vigenere table are then arranged according to its tabula key. During the 10 rounds of the decryption process, the ciphered key is first decrypted using the formula: Ki = (Ci + Ki) mod 95. After decrypting the key, the ciphered key and message are again interchanged through the following formulas: T=K; C=K; and K=CJ-1.

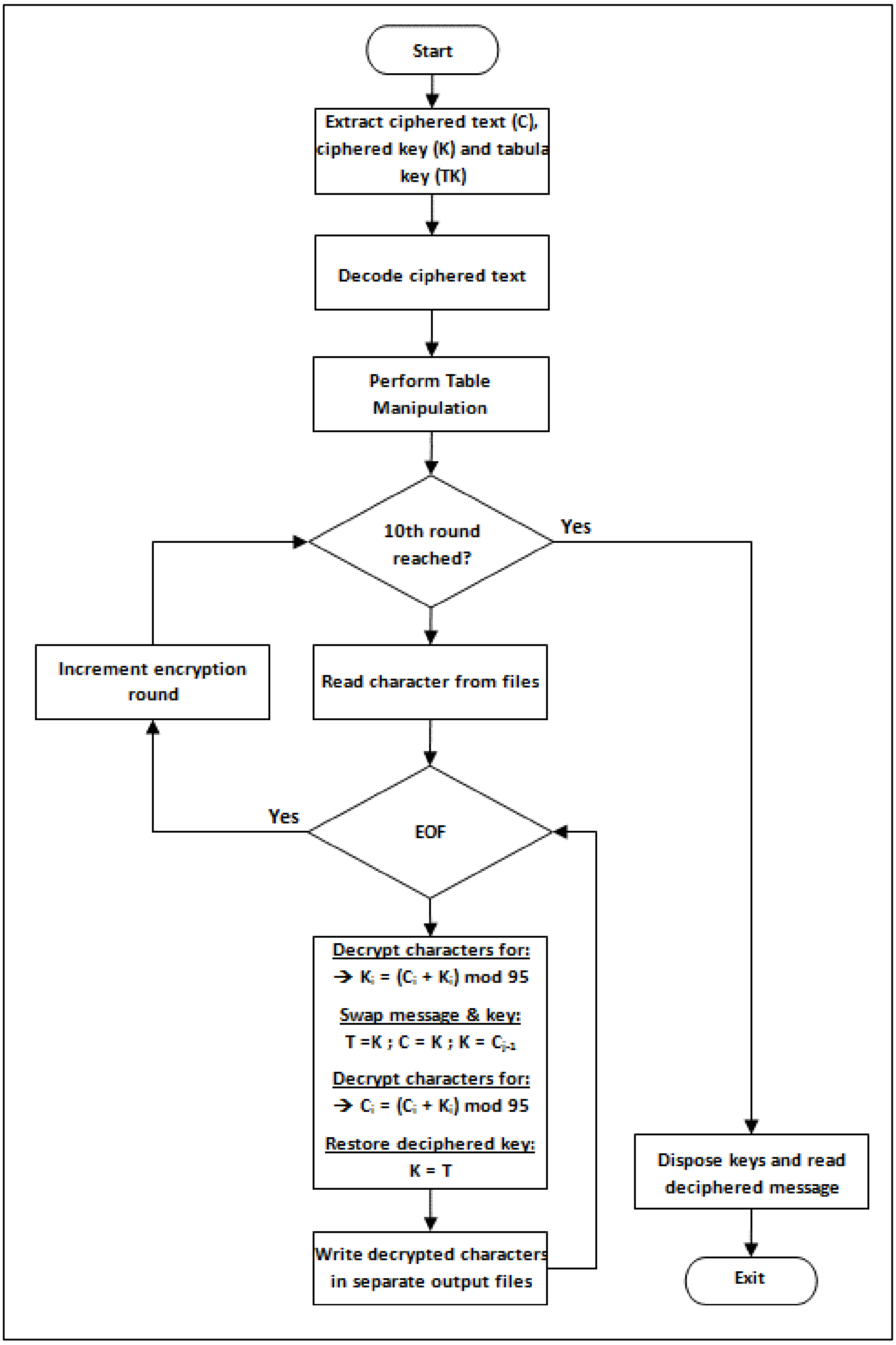
Start
Extract ciphered text (C), ciphered key (K) and tabula key (TK)
Decode ciphered text
Perform Table Manipulation
10th round reached?
Yes
Increment encryption round
Read character from files
Yes
EOF
Decrypt characters for:
→ K_i = (C_i + K_i) mod 95
Swap message & key:
T =K ; C = K ; K = C_{i-1}
Decrypt characters for:
→ C_i = (C_i + K_i) mod 95
Restore deciphered key:
K = T
Write decrypted characters in separate output files
Dispose keys and read deciphered message
Exit

Figure 3. Flowchart of the Decryption Process

The next process after this step is to decrypt the ciphered message using Ci = (Ci + Ki) mod 95. After the substitution process, the deciphered key is then reverted back to its original value. On the 10th iteration decryption round, the message including the key is revealed in its original form.

In general, every time the sender sends a message over a communication channel, the contents of the Vigenere table are freshly disarranged. It is important that the receiver receives the unaltered tabula key to unlock the right pattern of the table to be used in the decryption process. Both the tabula key and the ciphered keys are shared across the communication channel that is embedded within the encrypted file.

4. Results and Discussions

Every encryption technique has its own strengths and weaknesses. The analysis of these techniques based on several features is critically necessary. For this experimental study, the proponent used the Avalanche effect to analyze the performance and security of the proposed encryption algorithm.

Avalanche Effect refers to an enviable property of the cryptographic algorithms where, if the input is changed slightly, the output changes significantly. The following formula is used in this experiment:

$$Avalanche\ Effect = \frac{No.of\ Chars\ Changed\ in\ Ciphered\ Text}{Total\ No.\ of\ Chars\ in\ Ciphered\ Text} \times 100 \tag{2}$$

In this section, we have compared the Modified Vigenere Cryptosystem with the traditional Vigenere Cryptosystem using the Avalanche Effect as a performance metric. For all cases of the experiment, the input plain text was “We are discovered”, and the input cipher key was “break”. The system was set to generate a fixed hex value of 245A FF13 899C 7452 27BB D8C in which is applicable only to the modified version, but both versions will be using a 95x95 Vigenere matrix. Considering the different case scenarios, the input data for both versions of the encryption techniques were altered at the start, middle, and at the end of the plain text, cipher key, and the tabula key by changing only a single character from the string.

Table 3. Avalanche effect of altering the plain text by a single character

	Vigenere Encryption Techniques	
	Traditional Version	**Modified Version**
Plain Text	We are discovered	We are discovered
Cipher Key	break	break
Tabula Key	*(not applicable)*	245AFF13899C745227BBD8C
Encrypted Text	:XeC^HrJK_Fb\G^HW	SQN%.(},0LM+nII&8[\"dxQ
Case #1 - First Character of Plain Text Changed from “W” to “X”		
Modified Plain Text	**X**e are discovered	**X**e are discovered
Modified Enc. Text	**;**XeC^HrJK_Fb\G^HW	**Urp***.(},0LM+nII&8[\"dxQ
Avalanche Effect	5.88%	17.39%
Case #2 - Middle Character of Plain Text Changed from “i” to “u”		
Modified Plain Text	We are d**u**scovered	We are d**u**scovered
Modified Enc. Text	:XeC^HrJ**W**_Fb\G^HW	SQN%.(},**yJ**M+nII&8[\"dxQ
Avalanche Effect	5.88%	8.7%
Case #3 - Last Character of Plain Text Changed from “d” to “o”		
Modified Plain Text	We are discovere**o**	We are discovere**o**
Modified Enc. Text	:XeC^HrJK_Fb\G^H**b**	SQN%.(},0LM+nII&8[\"**n1S**
Avalanche Effect	5.88%	13.04%

The first three case scenarios were conducted by flipping a single character in the plain text on three specific locations. Specific portions of the modified encrypted texts were highlighted to indicate changes in their byte information.

In Table 3, the modified version of the Vigenere Cryptosystem has a higher avalanche effect over the traditional version on all three cases. Its avalanche effect ranges from 8.7% to 17.39%, that is, 2 to 4 characters were changed out of 23 characters.

The next case scenarios were conducted to determine the avalanche effect of both algorithms if the cipher key was to be modified.

Table 4 shows that the results have a higher avalanche effect when altering a single character on different locations of the cipher key. Still, the modified version has a higher result compared to the traditional version.

Table 4. Avalanche effect of altering the cipher key by a single character

	Vigenere Encryption Techniques	
	Traditional Version	**Modified Version**
Plain Text	We are discovered	We are discovered
Cipher Key	break	Break
Tabula Key	*(not available)*	245AFF13899C745227BBD8C
Encrypted Text	:XeC^HrJK_Fb\G^HW	SQN%.(},0LM+nII&8[\"dxQ
Case #4 - First Character of Cipher Key Changed from "b" to "g"		
Modified Cip. Key	**g**reak	**G**reak
Modified Enc.Text	**?**XeC^**M**rJK_**K**b\G^**M**W	**k3S*A**(},0LM+nII&8[\"**Bq"**
Avalanche Effect	23.53%	34.78%
Case #5 - First Character of Cipher Key Changed from "e" to "&"		
Modified Cip. Key	br**&**ak	br**&**ak
Modified Enc.Text	:X**&**C^Hr**j**K_Fb|G^HW	**3**QN%.(},**dF**M+nII&**J\H&**dxQ
Avalanche Effect	17.65%	30.43%
Case #6 - First Character of Cipher Key Changed from "k" to "%"		
Modified Cip. Key	brea%	brea%
Modified Enc.Text	:XeC**w**HrJK**x**Fb\G**w**HW	SQN%**b<z**,0LM+**z18%6Z**\"dxQ
Avalanche Effect	17.65%	39.13%

Table 5. Avalanche effect of altering the tabula key by a single character

Vigenere Encryption Technique: Modified Version	
Plain Text	We are discovered
Cipher Key	Break
Tabula Key	245AFF13899C745227BBD8C
Encrypted Text	SQN%.(},0LM+nII&8[\"dxQ
Case #7 - First Character of Tabula Key Changed from "2" to "F"	
Modified Tabula Key	**F**45AFF13899C745227BBD8C
Modified Encrypted Text	**t***N%**;'**},0**6@c0(**>&8**[$**"**40-**
Avalanche Effect	60.87%
Case #8 - Middle Character of Tabula Key Changed from "C" to "A"	
Modified Tabula Key	245AFF13899**A**745227BBD8C
Modified Encrypted Text	**5.g&z=**},**"cc**+n**gp*$%:#*-b**
Avalanche Effect	82.61%
Case #9 - Last Character of Tabula Key Changed from "C" to "F"	
Modified Tabula Key	245AFF13899C745227BBD8**F**
Modified Encrypted Text	**CU**N%**:\H&qdz"A`f&KK.%8dR**
Avalanche Effect	86.96%

The last three case scenarios as shown in Table 5 were carried out to determine the avalanche effect if the tabula key was altered. Since the traditional version doesn't have this type of key, the modified version was tested instead.

In this case scenario, the modified version has a better avalanche effect when a tabula key is modified. The result shows that by flipping a single character of the tabula key, more than 50% of the encrypted text will definitely change.

The graph shown in Figure 4 summarizes the overall result of the avalanche effect of both versions of the Vigenere encryption techniques used in this study.

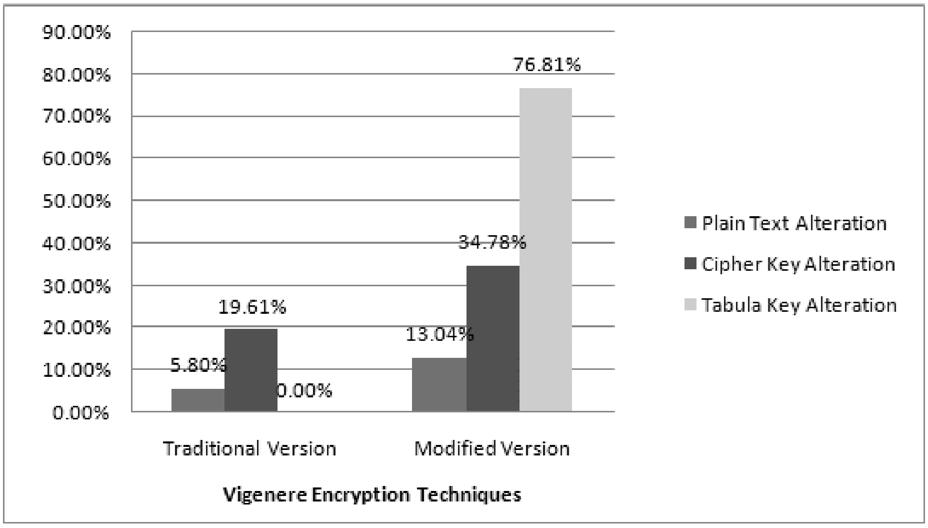

Figure 4. Overall Result of the Avalanche Effect

5. Conclusion and Future Extensions

In this paper, the proponent had designed and introduced a modified version of the Vigenere Cryptosystem using matrix manipulation over the Vigenere square and Base94 encoding scheme. The design also integrates a scheme that performs iterative rounds of substitution transformation of the message and key. The Modified Vigenere Cryptosystem showed a high avalanche effect as compared to the traditional Vigenere encryption algorithm.

This research opens up many possible avenues for future investigation. One particular area in this study could involve the combination of more efficient encryption algorithms such as an AES or RSA to increase the complexity and level of security. Another area for refinement that could be considered could be an integration of a compression scheme to reduce the size of the encrypted data being generated. Other encoding schemes could be considered to suit different applications or operating system platforms. The matrix manipulation scheme doesn't end here, there are better ways to shuffle the contents of the Vigenere Square with a better random value distribution.

References

[1] Aakash, J. K. Soni and B. Sharma, *"A. J. Cipher",* 2017 2nd International Conference on Telecommunication and Networks (TEL-NET), Noida, 2017, pp.1-6, doi: 10.1109/TEL-NET.2017.8343547.

[2] P. Thorsteinson, G. G. Ganesh, "*.NET Security and Cryptography (First Edition)*", USA, Prentice Hall, 2003, ISBN: 013100851X.

[3] F. Fahrianto, S. U. Masruroh, N. Z. Ando, "*Encrypted SMS application on Android with combination of Caesar cipher and Vigenere algorithm*", in 2014 International Conference on Cyber and IT Service Management (CITSM), November 3-6, 2014, South Tangerang, Indonesia, pp.31-33, doi: 10.1109/CITSM.2014.7042170.

[4] C. Danuputuri, T. Mantoro, M. Hardjianto, "*Data Security Using LSB Steganography and Vigenere Cipher in an Adnroid Environment*", in 2015 Fourth International Conference on Cyber Security, Cyber Warfare, and Digital Forensic (CyberSec), 2015, Jakarta, Indonesia, pp.22-27, doi: 10.1109/CyberSec.2015.14.

[5] Y. A. Gerhana, E. Insanudin, U. Syarifudin, M. R. Zulmi, "*Design of digital image application using vigenere cipher algorithm*", in 2016 4th International Conference on Cyber and IT Service Management, April 26-27, 2016, Bandung, Indonesia, pp.1-5, doi: 10.1109/CITSM.2016.7577571.

[6] K. Senthil, K. Prasanthi and R. Rajaram, "*A Modern Avatar of Julius Caesar and Vigenere Cipher*" in 2013 IEEE International Conference on Computational Intelligence and Computing Research, December 26-28, 2013, Enathi, India, pp.1-3, doi: 10.1109/ICCIC.2013.6724170.

[7] G. Singh, Supriya, "*Modified Vigenere Encryption Algorithm and Its Hybrid Implementation with Base64 and AES*", in 2013 2nd International Conference on Advanced Computing, Networking and Security, December 15-17, 2013, Mangalore, India, pp.232-237, doi: 10.1109/ADCONS.2013.33.

[8] W. Trappe, L. C. Washington, "*Introduction to Cryptographic with Coding Theory (Second Edition)*", USA, Prentice Hall, 2006, ISBN-13: 978-0131862395.