Cryptographic Algorithm Based on ASCII Conversions and a Radix Function

Y. Siva Rama Krishna

Abstract— Encryption is the process of encoding messages or information in such a way that only authorized users can read it. Cryptography to achieve a higher level of security. In this algorithm it becomes possible to hide the meaning of a message in unprintable characters by converting the ASCII value below 32. The main issue is to make the encrypted message undoubtedly unprintable using several times of ASCII conversions, Radix Function and a cyclic mathematical function. Dividing the original message into packets binary matrices are formed for each packet to produce the uprintable ecrypted message through the ASCII conversions, inverse cyclic mathematical and reverse radix functions are used to decrypt the unprintable encrypted message. The final message received from six times of encryption becomes an unprintable text through which the algorithm possess higher level of security without increasing the size of data or loosing of any data.

Key Words— Crytography, Encryption and Decryption, Higher Level of Security, Unprintable Encrypted Message, ASCII Conversions, Radix Base, Seed.

1 INTRODUCTION

Message authentication protects two parties who exchange messages from any third party. Several works have been done to develop a new cryptographic algorithm for a higher level of security. The cryptography is the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication and data origin authentication [1]. Also it means hidden writing, and it refers to the practice of using encryption to conceal text [3]. Cryptographic algorithms are used to encrypt and decrypt messages in a cryptographic system. Encryption transforms human redable plaintext into something unreadable, also known as ciphertext. The ciphertext is then decrypted to convert to the original plaintext, making it understandable to the authentic party. Among the available three modern securities offering techniques namely cryptography, stenography and watermarking, cryptography is the base to understand and also to implement ensuring a higher level of security in the real-time security systems. There are many approaches of cryptographic algorithem, most of them classified as symmetric key and asymmetric key cryptography. In symmetric key algorithm same key is used for both encryption of plaintext and decryption of ciphertext. Symmetric key encryption can use either stream ciphers or block ciphers. Asymmetric cryptography is cryptography in which a pair of keys is used to encrypt and decrypt a message. In this paper, we proposed a new cryptographic algorithm which follows a different methodology from the traditional symmetric-key cryptography, asymmetric-key cryptography or hashing function.

2 PREVIOUS WORKS

Several works have been done to develop a new cryptographic algorithm for a higher level of security. A new cryptographic algorithm for the real time applications was proposed by A. H. Omari, M.B. Al-Ksasbeh, E.R. Al Qutaish to improve the time for encryption and decryption of data of end-to =-end delay and to provide higher level of security[5]. Some researchers have developed a secure hybrid mode-based crytosystems which provides greater security level than that schemes based on a single hard problem. The enemy or adversary has to solve the two problems simultaneously which is unlikely to happen in order to read any secret message [6]. Some researchers have developed a new cryptosystem using multiple cryptographic assumptions which offers a greater security level than that schemes based on a single cryptographic assumptions [8]. S. Kumar, Addgarla, and Y. Babji made a comparative security study on symmetric key crptosystem based algorithms such as DES, TDES, IDEA, and AES [9]. A basic study on cryptography which is a solution for information security threats has been shown in [10].

3 PROPOSED ALGORITHM

3.1 Encryption Phase

In the encryption phase of the proposed algorithm, at first the input characters of the text to be encrypted are divided into several packetes of N characters taking inorder from the beginning character, where the value of N is 4 which may vary only for the last packet as the last packet contains the remaining characters. Its value may range from 1 to 4. For example, if a text consists of 15 characters, the first 4 characters constitue the first packet, the subsequent 4 characters constitue the second packet and the remaing 3 characters constitute the last packet. Secondly, a binary matrix p[N,8] is formed for each packet using 8-bit binary equivalent of the ASCII value of each character. The binary value of each ASCII of a packet is then accommodated row wise in the binary matrix. Thirdly, 8 new ASCII values denoted by NewASCII[i] for each packet is evaluated using the matrix P[N,8} taking the decimal equivalent of the bits belonging to each column of the matrix. In the example considered above, the NewASCII[1] for the first packet is the decimal equivalent of the bits from P[0,5] to P[0,0], where P[0,5} is the MSB and P[0,0} is the LSB.

The values of the NewASCII[i] range from 0 to 15 whose equivalent characters are unprintable. Now if we take the equivalent character for each NewASCII[i], then all the printable characters in the original data will become unprintable. Thus, this offers a better security making the ciphertext more secured.

Again, we are converted these NewASCII[i] values into Radix ASCII[i] by using radix base b. i.e. based on base b these decimal values are converted into another system. These values are stored in Radix ASCII[i]

A cyclic mathematical function has been used to encrypt once again the final ASCII values of the intermediate encrypted data. The mathematical function as shown below called cyclic because its output is rotated between 0 and 31.

$$FinalASCII[i]=(RadixASCII[i]+M)\%32, \tag{1}$$

Where, M value generated randomly

Finally the 8 FinalASCII[i], values are converted to their equivalent characters whose are undoubtedly unprintable so that the final ciphertext cannot shown at all. This encryption procees repeats for each packets of N characters for the original data. Then combining all packets in zig zag manner as a single, these packets are sent to the receiver.

3.2 Pseudocode of Encryption

The pseudocode of the encryption procedure using the parameters discussed above can be summarized as follows:

```
1. Input original message
2. Divide the original message into several packets of N
   characters
3. For each packet
   a. Convert the characters to their equivalent ASCII
   b. Create binary matrix P[N,8] with the binary value
      from the ASCII of the characters
   c. Calculate new ASCII as:
         Initialize i=0
         For (k=1 to 8)
            Increment i by 1
            NewASCII[i]=0
            For (j=0 to N-1)
               $NewASCII[i]+=(P[N-j,k])*2^{N-j-1}$
            End inner loop
         End outer loop
   d. Calculate RadixASCII as:
         For (i=1 to 8)
            RadixASCII[i]=radix(NewASCII[i], base)
         End loop
   e. Re-calculate each NewASCII[i] using the cyclic
      mathematical function:
         $FinalASCII[i]=(NewASCII[i]+M)\%32$
   f. Swap FinalASCII[i] values randomly using ran-
      domly generated Array
   g. Convert each FinalASCII[i] to its equivalent char-
      acter
4. End of encryption
```

3.3 Decryption Phase

In the decryption phase of the algorithm, at first the characters of the received unprintable ciphertext are converted to their equivalent ASCII. After which the inverse cyclic mathematcial function is applied to the ASCII as shown below:

$$DecASCII[i]=(ASCII[i]-M+32*R)\%32 \tag{2}$$

Then all the DecASCII[i] are divided into the same number of packets of 8 characters in order as done the encryption phase. Secondly, a binary matrix Q[N,8] is formed using the equivalent binary of the ASCII value in each packet, where the value of N is 4 which may vary only only for the last packet. Its value may range from 1 to 4. The binary value of each ASCII of a packet is then accommodated column wise in the binary matrix. Thirdly, the final N ASCII denoted by DesASCII-Final[i] for each packet is evaluated using the binary matrix Q[N,8] taking the decimal equivalent of the of the bits belonging to each row of the matrix. For the example considered in the above encryption phase, the DecASCII-Final[1] for the first packet is the decimal equivalent of the bits from P[0,0] to P[7,0], where P[0,0] is the MSB and P[7,0] is the LSB..

Finally, the N DecASCII-Final[i] values for each packet are converted to their equivalent characters whose are undoubtedly same as in the original message. This decryption process repeats for each packet of 8 characters of the encrypted data.

3.4 Pseudocode of Decryption

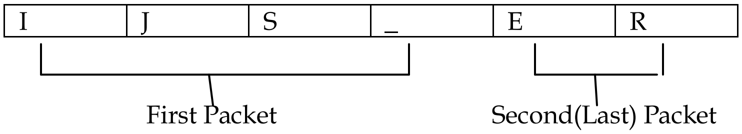
The pseudocode of the decryption procedure using the parameters discussed above can be summarized as follows:

```
1. Input encrypted message
2. Convert the characters to their equivalent ASCII
3. Arrange ASCII values regularly using randomly gen-
   erated Array
4. Calculate DecASCII[i] using inverse cyclic mathemat-
   ical function
      $DecASCII[i]=(ASCII[i]-M+32*R)\%32$
5. Calculate RevRadixASCII as:
      RevRadixASCII[i]=revRadix(DecASCII[i], base)
6. Divide the DecASCII[i] into several packetes of 8
   characters each
7. For each packet
      a. Create binary matrix Q[N,8] with binary
         values from the DecASCII
      b. Re-calculate new ASCII as:
            Initialize i=0
            For (j=1 to N)
               Increment i by 1
               DecASCII_Final[i]=0
               For (k=1 to 8)
                  $DecASCII\_Final[i]+=(Q[j,k])*2^{8-k}$
               End inner loop
            End outer loop
      c. Convert each DecASCII_Final[i] to its
         equivalent character
8. End of decryption
```

4. EXPLANATION WITH EXAMPLE

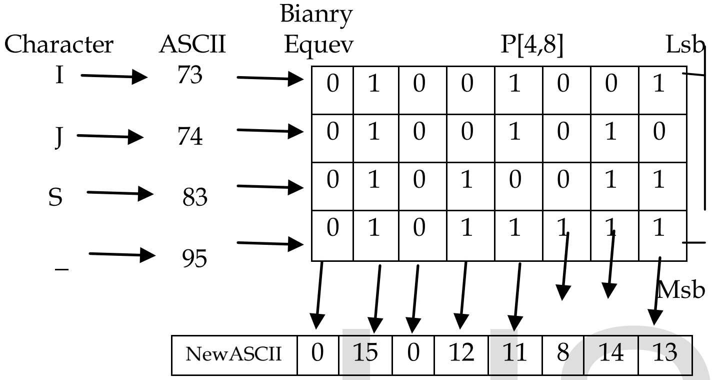
4.1 Encryption

As an example, let's consider that the user wants to encrypt and then conceal the message IJS_ER. According to the discussion the algorithm divides the input into 2 packets, where the first one contains the first 4 characters "IJS_" and the last contains the subsequent three characters "ER" as shown below:



Dividing the Original Message into 2 packets

As the explanation in the encryption phase the charactes of each packet are converted to their equivalent ASCII. Then using the 8-bit binary equivalent of the ASCII the first matrix P[4,8] is formed for the first packet as shown in Fig. 6 with the subsequent calculations to encrypt the characters of the packet.



Here we have taken base 6 all decimal values are converted into base 6 number system then we can get as follows:

RadixASCII	0	23	0	20	15	12	22	21

Then Final_ASCII is calculated using equation (1) where M values is randomly generated by using setSeed() method Here M value is 22. Then corresponding Final_ASCII is

FinalASCII	22	13	22	10	5	2	12	11

The encrypted character for Final_ASCII becomes unprintable because each of the ASCII is less than 32 which are unprintable

Encrypted charect for FinalASCII	.	.	.	.	.	.	.	.

Same procedure is applied for remaining packet, using the randomly generated array we can arrange the values of FinalASCII. Here randomly generated array size is 2 because we have taken only 2 packets, if we take 7 packets the array length becomes 7 and we can arrange those 7 packets randomly and send those to the receiver.

Finally, the original message is encrypted with a high security which becomes totally unprintable and hidden ensuring the main consent of the developed algorithm. The cipehrtext contains the 16 characters and the binary equivalent of the ASCII for each character is represented by only 4 bits. Since the plaintext contains only 8 characters and the binary equivalent of the ASCII for each character was represented by 8 bits, the ciphertext does not require extra bits just same as of the original.

4.2 Decryption

The ciphertext containing 16 characters are converted to their equivalent ASCII each of which is then converted to a new ASCII denoted by DecASCII using the inverse cyclic mathematical function. Then all the DecASCII are divided into 2 packets as discussed in the decryption phase each containing 8 ASCII. After which a Q[4,8] matrix for the first packet and a Q[2,8] matrix for the last packet are formed. From these matrices the original message is decrypted properly as shown below.

Encrypted Message

Convert the Encrypted Message each character to corresponding ASCII value then we can get

22	13	22	10	5	2	12	11	22	25	22	24	22	23	24	2

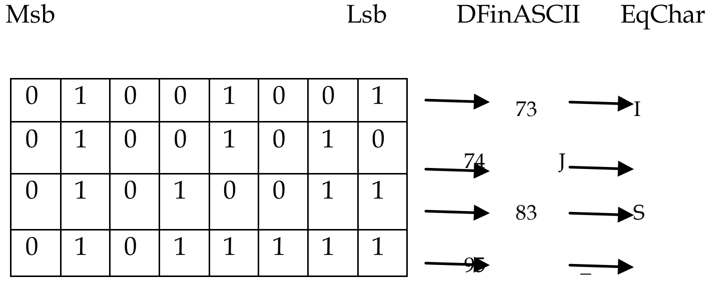
Using the equation (2) form DecASCII where R value is 1 because the generated M value is 22 it is below 32, if it is greater than 32 its value becomes 2, if its value greater than 64 its value is 3, and so on. Then the DecASCII is as follows:

0	23	0	20	15	12	22	21	0	3	0	2	0	1	2	1

Apply Reverse radix function on DecASCII array we can get DecRadixASCII. Convert the base 6 number system to decimal number system. We can get the following array values

0	15	0	12	11	8	14	13	0	3	0	2	0	1	2	1

Using the above array first we can decrypt the first packet the procedure is as follows:



Same procedure is applied for remaining packet, and combinig the characters we can get the entire decrypted message.

5. RESULT AND DISCUSSION

The result from the algorithm is very efficient with faster processing. A sample scenario of result of the developed algorithm implemented by java programming language is explained here. The plain text to be encrypted though the algorithm. Applying the encryption algorithm the ciphertext of unprintable characters is also shown. Java shows the boxes for unprintable characters. Thus, the ciphertext is a collection of boxes in which the original message is concealed. Again at the receiver, applying the decryption algorithm on the ciphertext original message is shown below. One point should be noted that only three extra bits are required to send with the cipher text to the receiver to denote the number of characters for the last packet whih is very negligible size used for efficient decryption. Again, the algorithm has developed in a new fashion which is different from the traditional symmetric-key cryptography, asymmetric-key cryptography or hashing function. In this algorithm, the ciphertext is produced through three times of encryption of the plaintext providing a higher level of security. Moreover, the input message to be encrypted and made hidden can be taken from a file like .txt, .doc, etc. as per the users's requirement. Similary, the final decrypted original message can be stored in the same file format.

However, the developed algorithm can be used in various security systems such as mobile communicaton system, E-mail communication system, cloud-based system, banking security system, network security system etc.

The secret code is $IJSER$(plain text)

@@@@@@@@@@   @@@@@@@@@(cipher text)

The secret code is $IJSER$

1. CONCLUSION

As cryptography is a big deal in all type security systems, this developed algorithm will be great part for same. We presented the algorithm based on ASCII conversions, radix function and a simple cyclic mathematical function. We emphasized mainly on the concealing of the encrypted message ensuring a better security. In future we try to implent some real-time security system using this algorithm.

REFERENCES

[1] E. Cole, R. Krutz and J.W. Conley, *Netowrk Security Bible*, Publishing Inc. 2005.

[2] Sidhpurwalahuzaifa A Brief History of Cryptography. [Online].

[3] A Menezes, V. Oosrschot and A. Vanstone, *Handbook on Applied Cryptography*, CRC Press Inc., NY, USA, 2000

[4] D. Stinson, *Cryptography Theory and Practice*, CRC Press Inc., NY, USA, 1995.

[5] A.H. Omari, B.M. Al-Kasabeh, R. E. A1-Qutaish, and M.I. muhairat, "A New Cryptographic Algorithm for the Real Time Application", *Proc. Of the 7th WSEAS International Conference on INFORMATTION SECURITY and PRIVACY (ISP)* , 33-38, 2008.

[6] E.S. Ismail and S. Baharudin, "Secure Hybrid Mode-Based Cryptosystem", *American Journal of Applied Sciences*, vol9, no 3, pp 289-292, 2012.

[7] E.S. Ismail and M.S. Jijazi, "New Cryptosystem Based on Factoring and Discrete Logarithm Problems", *Journal of Mathematics and Statistics*, vol 7, no 3, pp. 165-168, 2011.

[8] E.S. Ismail and M.S. Hijazi "New Cryptosystem using Multiple Cryptographic Assumptions", Journal of Computer Science, vol 7 no 12, pp. 1765-1769, 2011.

[9] S. Kumar, Addagarla, and Y. Babji, "A Comparative Security Study Review on Symmteric Key Cryptosystem Based Algorithms", International Journal of Computer Science and Mobile Computing, vol 2, no 7, pp 146-151, 2013.

[10] M.V. Kumar, "Cryptography- A solution for information security Threats" Golden Research Thoughts, vol. 2. No.1, 2013.