International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-9 Issue-2, December, 2019

Encryption Algorithm using Shuffled 2-Dimension Key

Sarika Y. Bonde, U. S. Bhadade

Abstract- ***Cryptographic algorithms are the fundamental element of security protocols and applications. They need to evolve to face the advance cyber security threats. This paper presents an encryption algorithm in which plaintext is encrypted using Shuffled 2-Dimension Key. Each time when a block is encrypted, the key is shuffled. Next time when a block is encrypted the key is different. Cipher text is more secured with shuffling 2-Dimension key as compared with same without shuffling 2-Dimension key. The results of 2-dimension array (shuffled and without shuffled) are compared with Advanced Encryption Standard (AES) algorithm. Same character is encrypted in different way as the key get changed due to shuffling.***

Keywords- Cryptography, encryption, decryption, AES, shuffle.

I. INTRODUCTION

Cryptography has been used from ancient times. But during the last two decades cryptography evolved tremendously. It is bedrock for pillars of modern world like e-commerce, communications and national security. In cryptography, the original message is called as plaintext. With the help of cryptographic algorithm plaintext is converted into a coded message. This process is known as an encryption and encrypted message is known as cipher text. The reverse of this process is known as decryption.

Cipher, a cryptographic algorithm, is the mathematical function used for encryption and decryption. Key-based algorithms are classified into two basic types: symmetric and asymmetric. In symmetric algorithms the encryption key and decryption key can be calculated from each other. In most of the symmetric algorithms, both keys are the same. These algorithms are also called as secret-key algorithms. The most popular symmetric key algorithms are DES, AES, IDEA, RC5, RC6 and Blowfish. Figure 1 shows the block diagram of symmetric key algorithm.

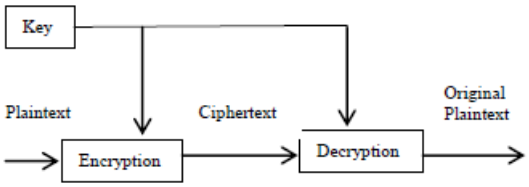

Fig.1: Symmetric key algorithm

Asymmetric algorithms are also known as public-key algorithms. These are designed such a way that the key used for encryption and decryption is different. RSA (Rivest Shamir Adlemam), Elliptic curve algorithm are the examples of asymmetric key algorithm [1, 2]. Figure 2 shows the block diagram of asymmetric key algorithm.

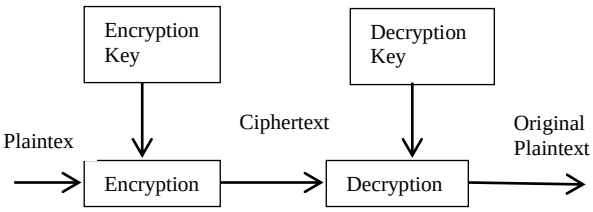

Fig.2: Asymmetric key algorithm

One of the cryptographic developments is shuffling. A shuffle consists of a permutation and re-encryption of a set of input ciphertexts. It is employed in cryptographic applications like multiparty computation, to build anonymous email, anonymous browsing.

A. Overview

The rest of this paper is organized in the following sections: section II shows related work. Section III describes the proposed work. Section IV discusses about the experimental evaluation, which is followed by conclusion.

II. RELATED WORK

Variety of methods for shuffling based cryptography had been provided by many researchers. Sarika Y.Bonde et al.[3] has proposed an encryption algorithm using 2-Dimension Key. 2-dimension array is the key, which will consist of random fashion ASCII text. Also result is compared with Advanced Encryption Standard (AES) algorithm. The use of 2-dimension array will provide security and saves effort of the data to be encrypted.V.B. Navya et al.[4] has propose shuffle mechanism which used 8 bytes of the key. The key is shuffled according to the shuffle order and new keys obtained are considered for sub keys generation. The novelty in key shuffling is by considering number of vowel's, 0's and 1's in the chunk.

Revised Manuscript Received on December 30, 2019.
* Correspondence Author
Sarika Y. Bonde, Research Scholar, Kavayitri Bahinabai Chaudhari North Maharashtra University Jalgaon, Maharashtra, India, sarika_apatil@rediffmail.com
Dr. U. S. Bhadade*, Professor and Head of Information and Technology department, Shram Sadhana Bombay Trust College of Engineering & Technology, Bambhori, Jalgaon, Maharashtra, India, umeshbhadade@rediffmail.com

Due to this it is very difficult for the others to get the information. Rishav Ray et.al [5] has proposes a method for hiding any encrypted secret message inside a text/ASCII or Microsoft word document file, by using the blank/white space characters of a cover file. The bits of each character of secret message file is inserted in place of eight randomly selected blank space characters of the cover file for hiding secret message inside any ASCII file. Related to positions of a shuffled offset matrix starting from a certain base address in cover file the randomly selected blank characters are read from cover file. Abdelfatah A. Tamimi et al. [6] has presented an encryption algorithm for audio files using a shuffling procedure. The enormous number of possible keys makes a brute-force attack on the algorithm impossible. Ernastuti [7] has proposed a asymmetry key encryption called as Perfect Shuffle Crypto Algorithm (PSCA). Transposition or permutation techniques in the crypto system are the classification of PSCA. For the linear plaintext length of N=2n, it will take O (N log N) to complete both encrypting plaintext and decrypting cipher text. S.Muthusundari et al. [8] has proposed an efficient encryption algorithm by an enhanced DShuffle Sorting technique. To improve the efficiency of text message the Bubble sort technique is replaced with an enhanced D-Shuffle Sorting Technique. Strength of this encryption algorithm is ASCII value translations and balanced tree representation of cipher text transmission. Richa Dubey et al.[9] has used a new combine technique of cryptanalysis and steganalysis using HTML file.. Only encryption/ decryption algorithm is not enough for complete confidentiality. Shuffle the text in such a manner that no one understands its true meaning. The use of text file is the cheapest and efficient way to hide these secret data as compare to many different media files. S. G. Rohini et.al [10] has proposed an substitution and shifting techniques based algorithm. Right shift or left shift is involves in shifting of the data. Finally concluded that the algorithm becomes complex by using shifting technique, which is hard to break and more secured. Sheela S. J. et al.[11] have considered modified Henon map and Sine map for chaos based text encryption algorithms. The encryption process can be divided into mainly four phases: Conversion of original message into equivalent ASCII codes, Shuffling of ASCII codes, Spiral scanning the ASCII codes and XOR operation. From the simulation results it is shown that the proposed algorithm maps the given plaintext into random ciphertext. Also the result of the chaos based algorithm is compared with RC4 algorithm. Jongho Won et al. [12] has propose a secure shuffling mechanism to enhance a white-box block cipher with dynamics in unmanned vehicle applications. Angelos Giakoumis et al.[13] has presents a Chaotic Random Number Generator (CRNG) based text message encryption technique. The CRNG creates a 5-bit random number for each message's character (from 0 [00000] to 31 [11111]). The program adds this 5-bit random number to the corresponding message's character to produce the ciphertext character (printable ASCII character). Abid Murtaza et al.[14] have presented an algorithm which is extremely simple and least complex and provides the higher level of security than popular symmetric algorithms such as DES and AES. Also shown that simplest software implementation of proposed algorithm is faster than some previously implemented conventional algorithms. Vinod Raghuvanshi et al.[15] has proposed Secure and Time Efficient Encryption Method (STEEM). STEEM provides high security of the encrypted text along with the time efficiency. STEEM Method is-the generated 'n' random number from input parameters will be formulated to create five random keys. Calculate length of message and multiplying it with key length. Divide the whole message in key length bits of 'k' blocks. Shuffle the data for each key bits length by using STEEM method.

Above revealed methods used shuffling mechanism for safety of data.

III. PROPOSED WORK

Since the security is provided with the help of 2-dimension array [3], here proposed shuffled 2-dimension array. For proposed shuffled 2-dimension array the different techniques are as follows:

Encryption:

Encryption method proposed here is based on encoding the plaintext. The encoding is done on plaintext using a special key. The method can be best explained using a example given below:

1. All possible plaintext characters are used in a key in special way (explained in key generation).
2. The encoding is started by reading the characters from the file to be encrypted.
3. The character is encoded by putting the row and column number of that corresponding character from the key which is treated as 2-dimension key.
4. Initially the row number and the column number are initialized to zero.
5. The character to be encoded is first searched in the 2-dimension key and the row and column numbers are updated accordingly, if the previous row number match with the current row number then only column number is used for encoding, else a special symbol is encoded with new row number followed by column number.
6. In this way if the character is found in the same row as that of previous then compression is achieved (normally 7 – bits are used to encode a plain ASCII text) whereas in our 2-dimension key only 5-bits are used to encode a character which results in saving of 2-bits per character.
7. The key is generated such that most probable characters are kept in each row.

Decryption:

The decryption process is very simple as compared to encryption process.

1. In this process the row and column number are again initialized to zero as it was done in encryption process.
2. The encoded bits are read and if it is seen that a special symbol was encounter than subsequent code is treated as row number followed by column number.
3. The character is then read from the 2-dimension key and is treated as plaintext.

Shuffling:

For strengthening the security each time the character is encoded, the position of that character in the 2-dimension key is shuffled.

1. The maximum number of characters in 2-dimension key is 128.
2. The shuffle counter is initialized to zero initially which is mod 128 counter.

3. The new character is formed by XORing current character with the character stored at the location pointed by shuffle counter.
4. The new column and row number are taken from the lower and upper nibble of the resultant character.
5. The character pointed by this new row and column is exchanged with the character pointed by the old row and column number.
6. In this way each time the character is encoded the dictionary get shuffled resulting in different code for the same character if that character appears multiple time in plaintext.

In this way additional security is provided for the encoding process.

IV. EXPERIMENTAL EVALUATION

As the use of text file is the cheapest and efficient way to hide these secret data as compare to many different media files [9] the proposed algorithms, Shuffle 2-dimension key and without Shuffle 2-dimension key also used text files size ranges from 1 KB to 200KB for effectively implementation. For execution Turbo C7 Simulator is used. Investigational results of Shuffle 2-dimension key, without Shuffle 2-dimension key are also compares with AES encryption algorithm. Encrypted text file size is the cipher text file which is obtained after encryption of plain text. The Decrypted text file size is the size of plain text file which is obtained after decryption of cipher text. Code1, Code2 and Code3 are the shuffled 2- dimension key .Code 4, Code5 and Code6 are the without shuffled 2- dimension key. All codes are generated by using 128 random numbers from 0 to 127.

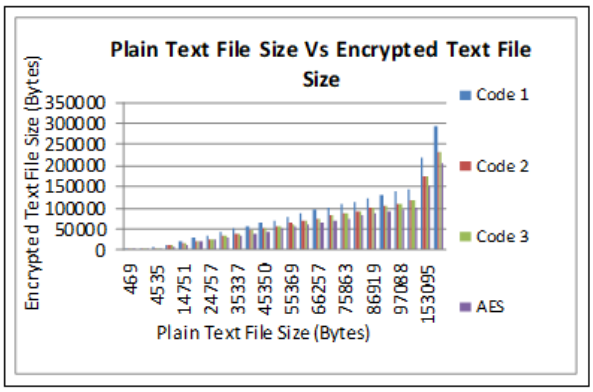

Fig.3: Plain Text File Size and Encrypted Text File Size comparison for
Shuffled 2- dimension key

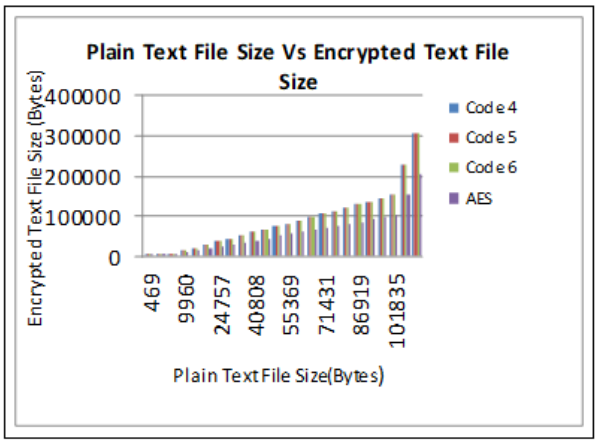

Fig.4: Plain Text File Size and Encrypted Text File Size comparison for
without Shuffled 2- dimension key

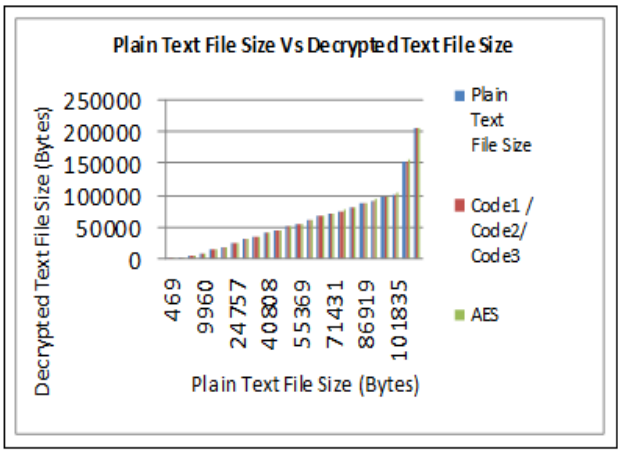

Fig.5: Plain Text File Size and Decrypted Text File Size comparison for
Shuffled 2- dimension key

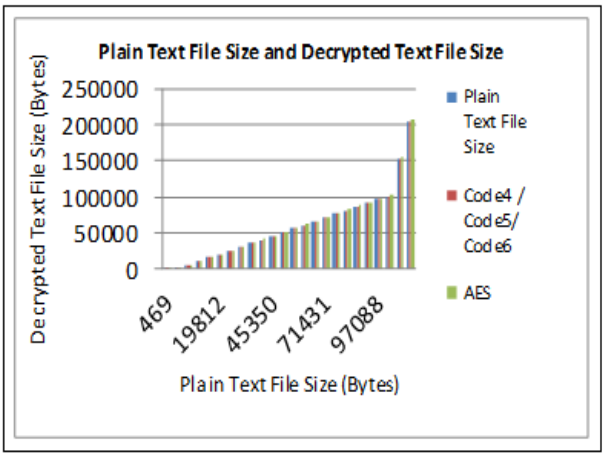

Fig.6: Plain Text File Size and Decrypted Text File Size comparison for without Shuffled 2-dimension key

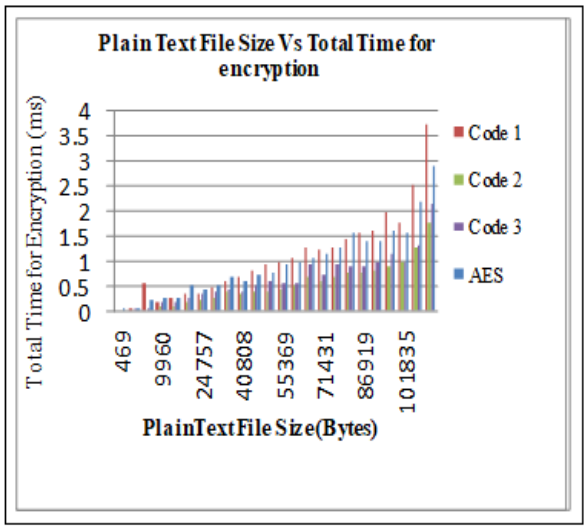

Fig.7: PlainText File Size and Total Time for Encryption comparison for Shuffled 2- dimension key

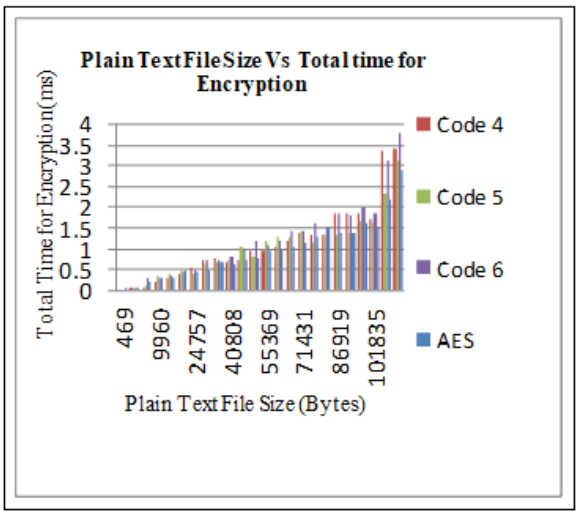

Fig.8:Plain Text File Size and Total Time for Encryption comparison for without Shuffled 2- dimension key

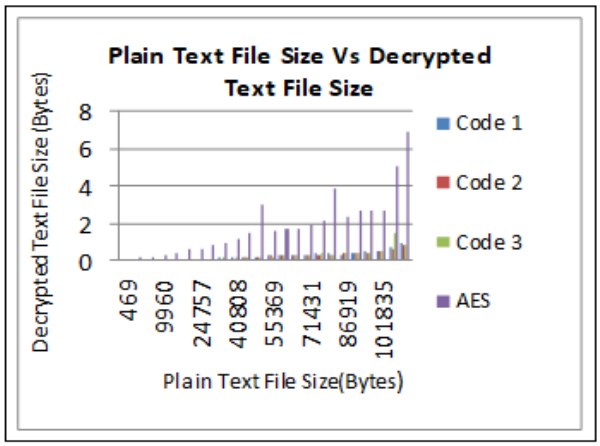

Fig.9: PlainText File Size and Total Time for Decryption comparison for Shuffled 2- dimension key

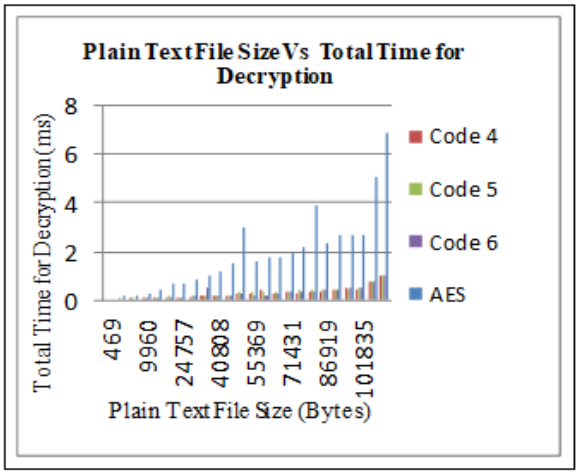

Fig.10: Plain Text File Size and Total Time for Decryption comparison for without Shuffled 2- dimension key

Experimental evaluation is as follows:

1. For shuffled 2- dimension key, plain text file size and encrypted text file size comparison is shown in figure 3. Results show that there is 2% decrement in encrypted text file size by simulating code 3.
2. For without shuffled 2- dimension key, plain text file size and encrypted text file size comparison is shown in figure 4. Results show that there is 33% decrement in encrypted text file size by simulating code 6.
3. For shuffled 2- dimension key, plain text file size and decrypted text file size comparison is shown in figure 5. Results show that decrypted text file size is exactly same as plain text file size by simulating code1/2/ 3. But by simulating AES algorithm there is 1% increment in decrypted text file size.
4. For without shuffled 2- dimension key, plain text file size and decrypted text file size comparison is shown in figure 6. Results show that decrypted text file size is exactly same as plain text file size by simulating code 4/5/6. But by simulating AES algorithm there is 32% decrement in decrypted text file size.
5. For shuffled 2- dimension key, plain text file size and total time for encryption comparison is shown in figure 7. Results show that least encryption time is required by simulating code 3. As compared with code1/2/3 there is 41% increment in time is required for encryption by simulating AES algorithm.
6. For without shuffled 2- dimension key, plain text file size and total time for encryption comparison is shown in figure 8. Results show that least encryption time is required by simulating code 6.While as compared with code 4/5/6 simulation of AES algorithm required 14% decrement in time for encryption.
7. For shuffled 2- dimension key, plain text file size and total time for decryption comparison is shown in figure 9. Results show that least decryption time is required by simulating code 3. While as compared with code 1/2/3 simulation of AES algorithm required 529% increment in time for decryption.
8. For without shuffled 2- dimension key, plain text file size and total time for decryption comparison is shown in figure 10. Results show that least decryption time is required by simulating code 6. While as compared with code 4/5/6 simulation of AES algorithm required 600% increment in time for decryption.

V. CONCLUSION

Shuffled 2-dimension array is the key used for encryption. For performance evaluation shuffled 2-dimension key and without shuffled 2-dimension key are used. Shuffled 2-dimension keys are Code1/ Code2/Code3 while Code4/ Code5/Code6 are without shuffled 2- dimension keys. With the help of 128 random numbers codes are produced. In 2-dimension key the number used are from 0 to 127.The shuffle counter is initialized to zero initially which is mod 128 counter. The new character is formed by XORing current character with the character stored at the location pointed by shuffle counter.

The new column and row number are taken from the lower and upper nibble of the resultant character. The character pointed by this new row and column is exchanged with the character pointed by the old row and column number. In this way each time the character is encoded the dictionary get shuffled resulting in different code for the same character if that character appears multiple time in plaintext. The shuffled 2-dimension key algorithm, without shuffled 2-dimension key algorithm and Advanced Encryption Standard (AES) algorithm are successfully perform for different text files size. Text files of 1 KB to 200KB are used. For simulation Turbo C7 Simulator is used. From experimental evaluation it is concluded that decrypted text file size is exactly same as plain text file size by using 2-dimension array but decrypted text file Size is 1% more as plain text file size by using AES algorithm. By using code 3 encrypted text file size is 2% less. By using code 3, decrypted text file size is 1% less as compared with AES algorithm. So, for shuffled 2-dimension key code 3 is best for encryption and decryption. By using without shuffled 2-dimension key encrypted as well as decrypted text file size is 33% less by using code 6. As compared with shuffled 2- dimension key, there is 41% increment in time for encryption by simulating AES algorithm. As compared with 2- dimension array, there is 564 % increment in time for decryption by simulating AES algorithm. The cipher text is more secured by using shuffled 2-dimension key as compared with same without shuffling 2-dimension key. Ciphertext becomes more secured when the key is shuffled each time after encrypting block of text.

REFERENCES

1. Bruce Schneir, “Applied Cryptography”, 2nd edition, John Wiley & Sons, 2007.
2. William Stallings, “Cryptography and Network Security”, Pearson Education, Fourth Edition, 2007.
3. Sarika Y. Bonde,Dr. U. S. Bhadade , “Encryption Algorithm using 2-Dimension Key for Information Security”, International Journal of Engineering and Advanced Technology (IJEAT),ISSN: 2249–8958, Volume-8, Issue-6, August 2019, pp. 4874- 4877.
4. V.B. Navya, R. Aparna, and G. Bhaskar, “Mobile Payment Security by Key Shuffle Mechanism in DES”, International Conference on Computational Intelligence and Information Technology (CIIT) Springer-Verlag Berlin Heidelberg, 2011,pp. 281-285.
5. Rishav Ray, Jeeyan Sanyal, Debanjan Das, Asoke Nath, “A new Challenge of hiding any encrypted secret message inside any Text/ASCII file or in MS word file: RJDA Algorithm”, IEEE International Conference on Communication Systems and Network Technologies, 2012, pp.889-893.
6. Abdelfatah A. Tamimi and Ayman M. Abdalla, “An Audio Shuffle-Encryption Algorithm”, Proceedings of the World Congress on Engineering and Computer Science(WCECS), Vol- I, San Francisco, USA, 22-24 October, 2014, pp.-1-4.
7. Ernastuti, “Perfect Shuffle Algorithm for Cryptography”, ARPN Journal of Engineering and Applied Sciences, ISSN 1819-6608, VOL. 9, NO. 12, December 2014, pp.-2383-2386.
8. S.Muthusundari I,R.M.Suresh, “An Enhanced D-Shuffle Sorting Algorithm for Secured Encryption Message to Represent in Tree”, IEEE International Conference on Advanced Communication Control and Computing Technologies (lCACCCT), 2014, pp.-1583-1588.
9. Richa Dubey, Apurva Saxena, Sunita Gond, “An Innovative Data Security Techniques Using Cryptography and Steganographic Techniques”, International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 6 (3),2175-2182,2015,pp.2175-2182.
10. S. G. Rohini, Ch. Jyothsna, Ch. Ramaiah, Sk. Madeena Sunny, “ASCII Based Symmetric Key Algorithm for Data Security”, International Journal of Pure and Applied Mathematics, Volume- 116 ,No. 5 2017,pp. 75-80.
11. Sheela S. J., Suresh K. V., Deepaknath Tandur, “Secured Text Communication using Chaotic Maps”, IEEE International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET), 16-18 February 2017,pp.1-6.

12. Jongho Won, Seung-Hyun Seo and Elisa Bertino, “A Secure Shuffling Mechanism for White-box Attack-resistant Unmanned Vehicles”, IEEE Transactions on Mobile Computing, Vol. 14, No. 8, August 2017, pp.1-17.
13. Angelos Giakoumis, Christos K. Volos, Jesus Manuel Munoz-Pacheco, Luz del Carmen Gomez-Pavon, Ioannis N. Stouboulos, and Ioannis M. Kyprianidis, “Text Encryption Device Based on a Chaotic Random Bit Generator”, IEEE 9th Latin American Symposium on Circuits & Systems (LASCAS) 2018, pp1- 4.
14. Abid Murtaza, Syed Jahanzeb Hussain Pirzada, Liu Jianwei. “A New Symmetric Key Encryption Algorithm With Higher Performance”, IEEE International Conference on Computing, Mathematics and Engineering Technologies – iCoMET 2019, pp.1-7.
15. Vinod Raghuvanshi, Pradeep Mewada and Praneet Saurabh, “Development of More Secure and Time Efficient Encryption Method”, Springer Nature Singapore Pte Ltd. 2019, pp.299-309.

AUTHORS PROFILE

Sarika Y. Bonde[1] has completed her B.E. from North Maharashtra University Jalgaon, (M.S). M.E. from Dr.Babasaheb Ambedkar Marathwada University, Aurangabad(M.S.).Currently she is research scholar from K.avayitri Bahinabai Chaudhari North Maharashtra University Jalgaon, (M.S) INDIA. In National/International Conference and Journal she has published/presented 13 papers. Her area of interest is Object Oriented Programming, Digital Signal Processing and Cryptography. She is Life Member of ISTE.

Prof.Dr. U. S. Bhadade[2] has completed his B.E from Pune University (M.S.), M.E. from Amravati University (M.S) and Ph.D. from M.S. University Baroda. Currently he is Professor and Head of Information and Technology department at Shram Sadhana Bombay Trust College of Engineering & Technology Bambhori, Jalgaon, (M.S) INDIA. In National/International Conference and Journal he has published/presented 51 papers. His area of interest is Microprocessor, Text Compression, Software Engineering and Computer Network. He is Board Member of IJETT (Journal). He is Life Member of IETE, ISTE and IJERIA.