iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** VIII **Month of publication:** August 2015

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

The PolyVernam Cipher

Divya R[#1], Anita Madona M[#2]
[#]*Computer Science, Auxilium College, and Katpadi*

Abstract— Cryptography encryption is an effective way to achieve the security of data. The encryption is to hide the data in a way that an attacker cannot hack the data. The main purpose of encryption is to hide the data from the unauthorized parties. The symmetric key cryptographic is called the conventional/private-key/single-key cryptography. The sender and receiver can share a common key for encrypting and decrypting the data. An important distinction in the symmetric cryptographic algorithms is between stream and block ciphers. Stream ciphers convert one symbol of plaintext directly into a symbol of cipher text. Block ciphers encrypt a group of plaintext symbols as one block. The symmetric cipher use similar keys. For encryption, the plaintext is in the input side, whereas for decryption, the cipher text is in the output side. The symmetric key algorithm is much faster than the asymmetric algorithm. Symmetric key ciphers can be used as primitives to construct various cryptographic mechanisms. It can be composed of producing a stronger cipher. This is the advantage of the symmetric cipher. The disadvantage of the symmetric key is shared key systems; however, both parties know the secret key. In polyalphabetic cipher, multiple alphabets are used. To facilitate encryption, all the alphabets are usually written in a large table. The advantage of polyalphabetic cipher is that they make a frequency analysis more difficult. The polyVernam cipher is a one of the method of encryption. This cipher is not easy to understand and implement, so it often appears to be unbreakable. A new approach is polyVernam cipher is designed by combining of polyalphabetic cipher and Vernam cipher to increase the security in the symmetric cipher. A polyalphabetic cipher is based on substitution cipher. The polyalphabetic decrypt value can be substituted in the polyVernam cipher; while decrypting the value, it gives a new encrypted value. Using the same key can get the original message from the encrypted value. The encrypted message is applying by XOR operation. And finally, a receiver can get the message more securely.

Keywords— Asymmetric cryptography, one-time pad, polyVernam cipher, symmetric crytograpy, symmentric stream cipher.

I. BACKGROUND

Cryptography has a main important role in security of data. Security is the important factor in public network. The objective of cryptography is not only to provide the security. The security protect implemented by encryption/decryption. Cryptography is the method that allows information to send in secure from such a way that only the sender and the receiver can retrieve that information. The symmetric key cryptography uses the same key for encryption and decryption processes. In the asymmetric key cryptography, two separate keys are used: one for encryption process and another one for decryption process.

II. PROBLEM STATEMENT

In the cryptography algorithms, there are some parameters where improvement necessary. These parameters are efficiency, execution time, throughput value, time complexity, and many more. All the algorithms use complicated keys or complicated algorithms for encryption and decryption. A key length is an important for designing for any algorithm, because a larger key length will cause slow execution and will reduce the performance of algorithm, whereas a smaller key length may result in poor security in the process. Security efficiency is more important of in any algorithm. The level of security of all algorithms is dependent on either the number of iterations or the length of keys. No single algorithm is sufficient for this purpose. To overcome this problem, ployVernam based on symmetric key cryptography is to improve the security and efficiency of the data. The algorithm targets the nodes of the network, where it is operating on, and the fact that there can be the nodes of varied architecture and processing powers, which are inter-linked. This algorithm considers the amount of processing power and memory available at every node, thereby ensuring seamless operation, which is both compatible and efficient enough.

III. APPROACH CHOSEN TO SOLVE THE PROBLEM

The polyVernam cipher is a symmetric stream cipher, which is aimed at providing a fast and efficient way of securely transmitting data and address of the fundamental problems of any stream ciphers —matching the key length with the input stream size without repetition in the key stream. This algorithm is especially applicable to the networks where fast and secures encryption and decryption of data is critical. In polyVernam cipher, a repetition of key letter again makes it less secure. As in above taken example after 7 key, alphabets again repeat first letter of keyword. Thus, an analyst can easily detect the repeated sequence of same cipher

text and make the assumption that the keyword is of the same length. It simply requires a key of length 56 bits to encrypt virtually any size data file. The operation to combine the message with the key is more computationally intensive, resulting in over 100 operations per 32 bit word. Most computers are not able to generate really random keys.

A. Key Generation Process

Key generation process is the depending on the encryption and decryption process. The key generation process involved the cryptography algorithm. In 256-bit key, the following algorithm is used for key scheduling.

for *i* from 0 to 255

S[*i*] := *i*

endfor

j := 0

for *i* from 0 to 255

j := (*j* + S[*i*] + key[imodkeylength]) mod 256

swap values of S[*i*] and S[*j*]

endfor

The second phase uses the following algorithm

i := 0

j := 0

while

i := (*i* + 1) mod 256

j := (*j* + S[*i*]) mod 256

swap values of S[*i*] and S[*j*]

K := S[(S[*i*] + S[*j*]) mod 256]

output *K*

endwhile

B. Key Generation Algorithm

Step 1: A 256 × 256 matrix is generated where each column has numbers from 0–25 indicating the 256 ASCII characters.

Step 2: Each of the columns is randomly shuffled producing a matrix of permuted columns each having values from zero, as shown in Fig. 1.

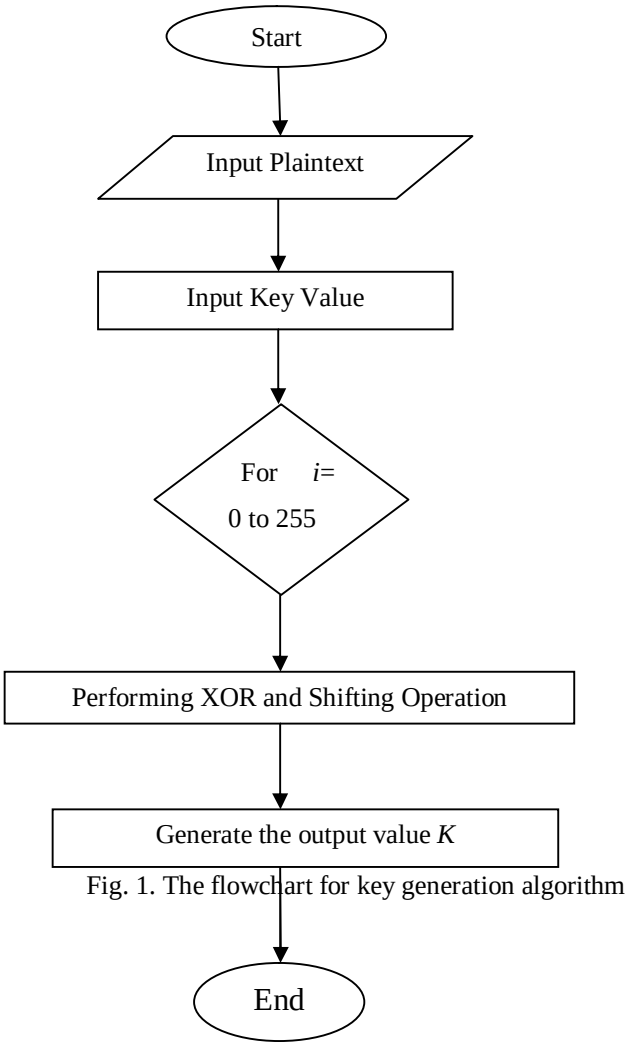


Fig. 1. The flowchart for key generation algorithm

C. Encryption Process

Encryption is the process of converting plaintext into cipher text. The encryption process is involved. Based on a polyVernam cipher, cryptography algorithm is stated here.

D. Encryption Algorithm

Step 1: A randomly chosen point in the matrix say (*x*,*y*). Each point has eight neighboring cells, hence eight directions for a point to move in.

Step 2: We substitute these directions with numbers from 0 to 7. The starting coordinates are saved as the direction.

Direction substitution table:

0	Top
1	Top right
2	Right
3	Bottom right
4	Bottom
5	Bottom left

6 Left

7 Top left

Step 3: The input Stream is read character by character, and the position selected is moved along the direction chosen in the matrix.
Step 4: Each character encountered in the traversal of the matrix is XOR with the character in the input stream and the position is incremented in the direction.

Step 5: At a particular iteration depending on the length of the input stream, the read position of the matrix reaches the boundaries of the current matrix. The next generation key is now generated.

Step 6: The next generation of the key is generated by utilizing the Rjindael forward S-Box transformation on the current matrix. Given the properties of the transform, a new matrix is obtained, which is revertible back to the previous version, as shown in Fig. 2.

Step 7: This is the next generation matrix, which is used as a crypto matrix for the following characters from the input stream.

Step 8: The starting point of this matrix is the same point as the place where the last read position collided with the boundaries of the last matrix.

Step 9: The same method is followed till another collision happens with the current matrix's boundaries, and the aforementioned steps are repeated until the entire message is encrypted.

Step 10: The key generated for this particular encryption session is the initial 256 × 256 permuted matrix, the initial starting point of the matrix then followed a tuple having the directions the encoder took while encrypting the data

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
10	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
20	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
30	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
40	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
50	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
60	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
70	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
80	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
90	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a0	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b0	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c0	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d0	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e0	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f0	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Fig. 2. The Rijndael S-box

E. Decryption Process

The decryption process involves grouping the cipher text that converts into the plaintext, and it involved in the polyVernam based on cryptography is stated below.

F. Decryption Algorithm

Step 1: The key is read, and the 256 × 256 matrix is reconstructed followed by the starting position and the direction.

Step 2: The read pointer is placed on that position and read moved along the matrix in that direction XORing the current input

character with the current matrix character.

Step 3: Just like the encryption, the decryption works by following the matrix data along the particular direction until a collision takes place at the boundaries, and the subsequent generation of matrices is produced, which decrypts the data.

IV. PRINCIPLE AND OPERATION OF POLYVERNAM CIPHER

A. Principle of PolyVernam Cipher

The original version of the cipher worked with only the base English alphabet of 26 letters. Numbers 0 through 25 are assigned to these letters (A to Z). For the *i*th character of classified data D_i by the key K_i (key is also composed only of the characters of the alphabet), the character of encrypted data C_i is determined as follows.

$C_i = (D_i + K_i) \bmod 26$

Decryption is performed by the inverse operation

$D_i = (26 + C_i - K_i) \bmod 26$

For example, the word "AGE" would be encrypted with the key "UHK", as shown in Table 1.

TABLE I

THE PRINCIPLE OF POLYVERNAM CIPHER

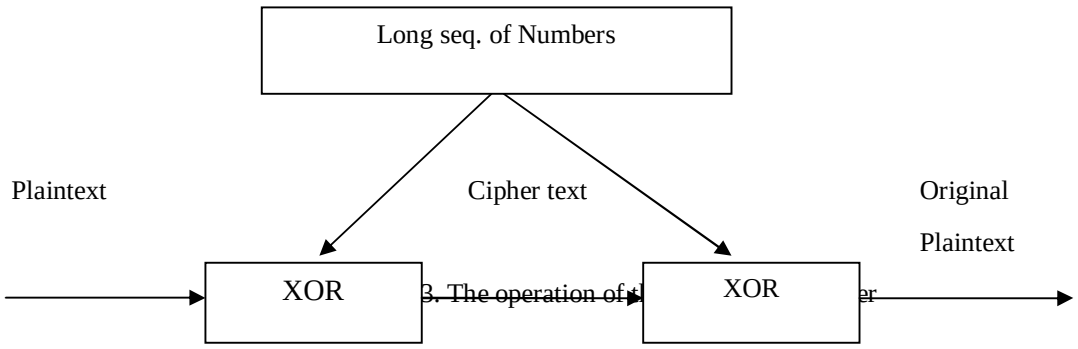
	Characters			Numbers		
Data	A	G	E	0	6	4
Key	U	H	K	20	7	10
Encryption	U	N	O	20	13	14
Decryption	A	G	E	0	6	4

An improved version of this cipher works with a binary data representation. The individual bits of data in the binary form are encrypted by the XOR operations with individual bits of the key. The advantage was the ability to machine processing of the cipher.

B. Operation of PolyVernam Cipher

The polyVernam was described as *impossible of translation* in the respected journal, "*Scientific American*". The alphabet cipher provides a good description of how to use a table for encryption and decryption using arbitrary keywords, but here is an alternate description (Fig. 3).

Plaintext + Key = Ciphertext ⇒ Ciphertext + Key = Plaintext.



3. The operation of t… …er

The following is the example for the polyVernam cipher.

1) The encipher chooses a plaintext: VIGENERE.
2) The encipher chooses a keyword and repeats it to become the length of the plaintext, e.g., the keyword, "CIPH": CIPHCIPH.
3) To encipher letter L1 of the plaintext, the encipher creates a new alphabet where in A is shifted to letter L1 of the cipher text, B is shifted to the next letter, and so on.

ABCDEFGHIJKLMNOPQRSTUVWXYZ

CDEFGHIJKLMNOPQRSTUVWXYZAB

4) The encipher finds the letter that corresponds to L1 in the substitution alphabet. This is now L1 of the plaintext: V⇒X.
5) This is repeated for each letter in the plaintext and its corresponding letter in the key: VIGENERE + CIPHCIPH ⇒XQVLPMGL.

C. One Time Pad with PolyVernam Cipher

The polyVernam cipher gives perfect secrecy depends on the assumption that each pad is equally likely. If the pad is used to encipher more than one message, this is no longer true, and the message may be discovered. It is important that a pad once used is discarded. That is the reason for the name one-time-pad, also known OTP. If this warning is not heeded, the two cipher texts can be subtracted, thus eliminating the pad. What is left is the difference of messages, which has a distribution reflecting back on the possibility of choice of pad. This has been known to completely break the cipher.

D. Unique Key Generation and the PolyVernam Ciphering

The randomness of the algorithm depends on the fact that for each successful decryption, the key is used to be destroyed, and never to be used again. A central database of the unique identification of the keys is to be maintained, which keeps a track of the keys generated. The method used to achieve that this is via the SHA 256 finger printing .The initial 256 × 256 matrix is line raised into a (256^2) length string and is used as the input for the SHA 256. A fixed length 256-bit hash is obtained, which is then inserted into the key database provided its unique or else, another initial key is generated, which is again verified through the same process until a unique SHA 256-bit finger print is obtained. Such a method is highly reliable and secure in a centralized network system.

V. RESULT

As the polyVernam cipher uses a random key shuffling mechanism including a random direction mechanism, a simple brute force attack on this cipher would take too long to crack. The shuffling of the key itself results in 256^{256} possibilities of the key. Furthermore, the random selection of the start point has ((254 × 254 × 8) + (4 × 254 × 5) + (4 × 3)) possibilities, and at every collision, there are a minimum of three and a maximum of five possibilities. Therefore, in the worst case, for *n* collisions, there are *n* × 5 possibilities. Thus, in total for a worst-case brute force scenario, it would require

((256^{256}) × ((254 × 254 × 8) + (4 × 254 × 5) + (4 × 3)) × 5 × *n*) hits to key, which is the key right. Here, *n* is the number of collisions, which is dynamic.

Hence, the polyVernam cipher can be certified as immune to brute force as this cipher is mainly targeted for the networks and high-speed networks like vehicular ad hoc networks, and in such scenarios, a brute force would take too long to crack the cipher and thus would render the cracked key to be useless.

A. Result Comparison in Tabular Form

A computer simplifies the process because the message is encoded in binary. Each character is represented internally by a computer as a unique combination of zeros and ones called bits, for example, the letter 'b' is composed of the eight bits '1100010'. This binary number is 98 in decimal.

To encrypt the message, each bit of each letter in the plaintext is combined with the corresponding letters' bit in the pad in sequence using a transformation called the bitwise exclusive or (abbreviated to XOR). The binary process is shown in Table 2.

TABLE II
THE BINARY PROCESS

Input bits		Output bit
Message	Pad	
0	0	0
0	1	1
1	0	1
1	1	0

B. Secure Cryptosystem for PolyVernam Cipher

All the methods of encryption ever devised, only one has been theoretically proved to be completely secure. It is called the polyVernam cipher or one-time pad. The worth of all other ciphers is based on the computational security. If a cipher is computationally secure, this means that the probability of cracking the encryption key using the current computational technology and algorithms within a reasonable time is supposedly extremely small, yet not impossible. In theory, every cryptographic algorithm, except for the polyVernam cipher, can be broken giving enough cipher text and time, for example, the public key cryptosystems, such as pretty good privacy.

VI. CONCLUSIONS

The original principle of polyVernam cipher proposed and modified to work with the keys, which were very complicated its practical use. When combined with modern polyalphabetic cipher algorithms, it is possible to encrypt data by using the key generation algorithm procedures and can also be repeatedly used as a simple password. Therefore, the strength of ciphers is always directly proportional to the strength of the chosen password. Implementation of this procedure is a programmatically very simple. The speed of encryption and decryption of data is mostly dependent on the speed of calculation of a hash code, i.e., on the selected hash algorithm. The polyVernam cipher has reached the maximum level of security that ciphers can provide us on the traditional computing systems. This cipher is not meant to be comparable with other ciphers. In such cases, the unique features of this cipher, such as cache optimization and redundancy check, would provide a tremendous speed boost to the encryption and the decryption process and is, therefore, ideal in situation where the speed is of essence in networks. The experimental results show that the proposed algorithm is very efficient and secured. In the future, all the above proposed algorithm is a simple and straightforward but intrinsically strong and compact approach to cryptography using essence of operations. It provides some time same or even better level of security using minimal time of complexity. This paper gives a step-by-step insight to three algorithms: 1) polyVernam cipher, 2) polyVernam tableau, and 3) data encryption. In the future, the works for algorithms, such as AES, RSA, and Two-fish, can be added that to simplify the complex algorithms. Second, more languages can be supported for the existing work as well as for any new work to be developed. This work runs as a stand-alone application and could later be converted into a web application, so that the user need not to download the application and instead of run it from the browser. The most striking feature of polyVernam cipher is that it is a far faster cryptographic algorithm than another, and it provides maximum security possible in the cipher text in the cost efficient manner.

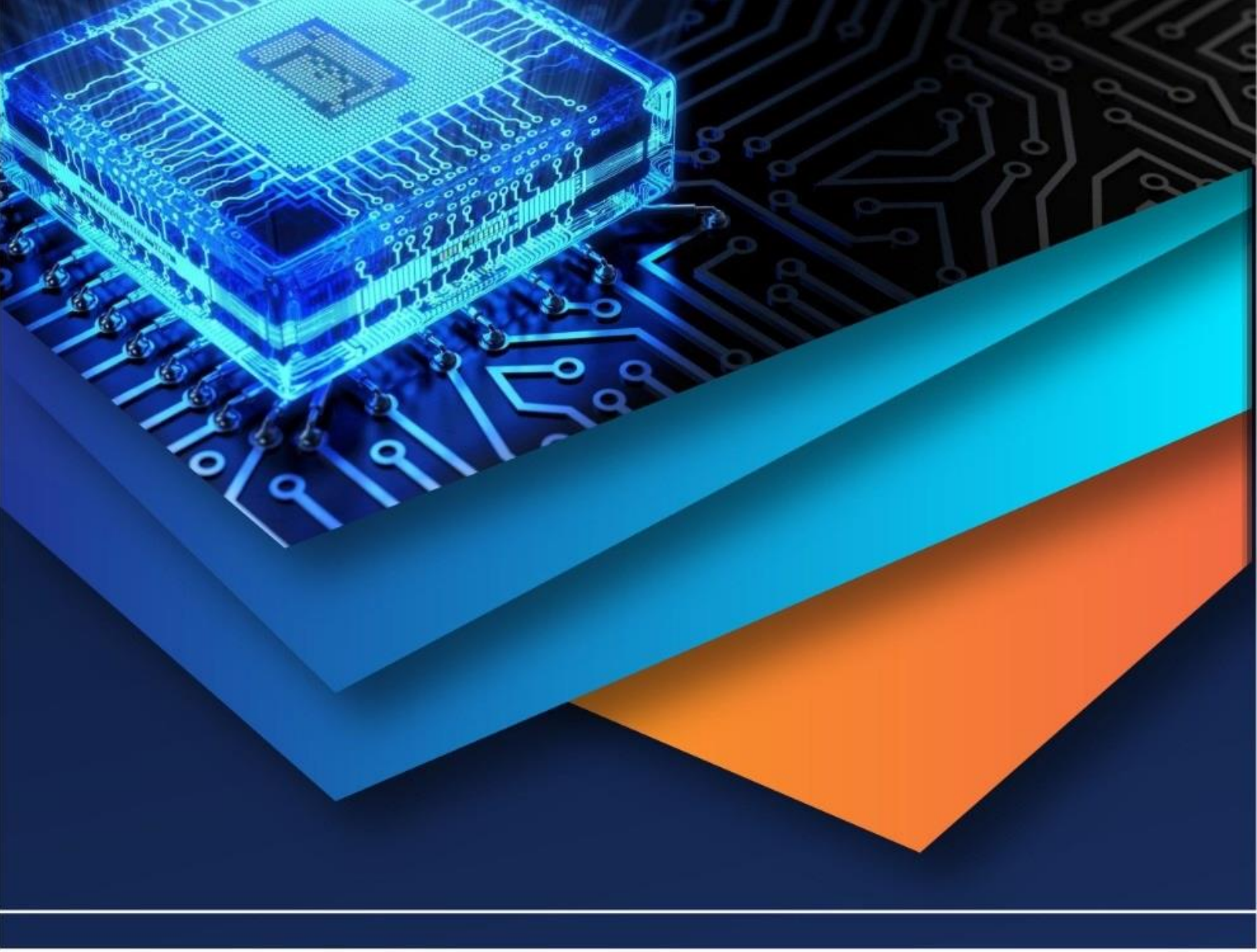
VII. ACKNOWLEDGMENT

The authors would like to thank the anonymous referees for their valuable comments, which greatly improved the readability of this paper. R. Divya would also like to thank M. A. Madona, Assistant Professor, Department of Computer Science, who guided for my work, and also express my whole hearted thanks to my parents and friends for their encouragements to bring this work to a successful completion.

REFERENCES

[1] Vernam, Gilbert S. (1926), "Cipher Printing Telegraph Systems for Secret Wire and Radio Telegraphic Communications", Journal of the IEEE 55:109–115.
[2] Alberti, Leon Battista (1997), “A Treatise on Ciphers, trans. A. Zaccagnini Foreword by David Kahn”, Torino Galimberti.

[3] Klein, Melville, Securing Record Communications: The TSEC/KW-26, retrieved 2012/04–12.
[4] Nicolas Courtois, Josef Pieprzyk, "Cryptanalysis of Block Ciphers with Overdefined Systems of Equations". , ASIACRYPT 2002.
[5] Joan Daemen, Vincent Rijmen, "The Design of Rijndael: AES – The Advanced Encryption Standard." Springer, 2002, ISBN 3-540-42580-2.
[6] Jawahar Thakur, Nagesh Kumar, "DES, AES and Blowfish: Symmetric Key Cryptography Algorithms Simulation Based Performance Analysis", International Journal of Emerging Technology and Advanced Engineering, December 2011.
[7] Hamm Eken, "Security Threats and Solutions in Computing" based on Cryptography Function," International Journal of Engineering and Innovative Technology (IJEIT), December 2011.
[8] Soulioti, V., Bakopoulos, Y., Kouremenos, S., Vrettaros, Y., Nikolopoulos, S., Drigas, A., Stream Ciphers created by a Discrete Dynamic System for Application in the Internet, WSEAS Transactions on Communications, Issue 2, Volume 3, April 2004, ISSN 1109–2742.
[9] Pierre, L., Richard, S., TestU01: A C Library for Empirical Testing of Random Number Generators Universities de Montréal: ACM Trans, 2007, Math. softw. 33, 4, Article 22.
[10] Strnadová, V., Interpersonální komunikace, Hradec Králové: Gaudeamus, 2011, 543 p., ISBN 978-80-7435-157-0.
[11] Chiunhsiun, L., Ching-Hung, S., Hsuan, S., H., Kuo-Chin, F., "Cryptography Using Polyalphabetic Technique Method of PolyVernam Cipher". NAUN International Journal of Circuits, Systems and Signal Processing, Issue 6, vol. 5, 2011, pp. 565–580, ISSN 1998–4464.
[12] Applied Cryptography: Protocols Algorithms and Source Code in C. Bruce Schiener. Second Edition, Oct 2006.ISBN-10: 0471117099.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY