Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 78 (2016) 217 – 223

Procedia Computer Science

International Conference on Information Security and Privacy, 11-12 December 2015, Nagpur, INDIA

Triplicative Cipher Technique

Rahul Johari[a], Harshit Bhatia[b], Shiwani Singh, Meena Chauhan[d]

[a]USICT, GGSIPU, Dwarka Sector 16C, New Delhi, India - 110078

Abstract

Cryptography, the study of "secret writing", is used today to protect the valuable data on internet and extranet. In computer networks assets like servers, user accounts, password and many more are critical and need protection. Cryptography serves the sole purpose of information security i.e., "to protect assets". Main aim is to achieve the three security goals- confidentiality, integrity and authenticity. To achieve these securities this paper proposes a robust encryption technique by considering alphabetical, numerical and alphanumerical data. It is purely a mathematical cryptography technique which comprises AES standard consisting of mathematical operations like addition, subtraction, multiplication and conversion methods.

Peer-review under responsibility of organizing committee of the ICISP2015

Keywords: Cryptography: Encryption: Decryption: Information Security: AES

1. Introduction

Cryptography and encryption have been used for many years for providing secure data transfer or sharing. This new method of information exchanging information has caused a tremendous need of concealing the data. As technology is growing faster and becoming more efficient, a thorough understanding about cryptography and encryption will serve people to develop better ways of concealing the valuable information. By securing the data it doesn't mean that cryptography should only provide confidentiality but it should also provide data integrity, non-repudiation and authenticity.

Corresponding Author: Tel: +91-7838596433
E-mail Address*: rahul@ipu.ac.in , harshit.usit.022164@ipu.ac.in, singh.shiwani7@gmail.com, meena.chauhan2028@gmail.com*

Peer-review under responsibility of organizing committee of the ICISP2015
doi:10.1016/j.procs.2016.02.036

1.1. Proposed System

The paper had represented a well secured encryption technique in which some attempts have been made to hide message, or the meaning thereof, in some medium. The technique presented in the paper provides secret sharing to support privacy-preserving, data outsourcing. Basically, the proposed technique is a combination of traditional symmetric key ciphers techniques that had used three special keys for both encryption and decryption and that's why named as- "Triplicative Cipher Technique" and it is a substitution cipher technique that can be categorized as either monoalphabetic cipher or polyalphabetic ciphers.

The technique proposed was a symmetric key cipher. Three prime numbers were selected which had served as the keys for the cipher. Mathematical operations multiplication, addition and subtraction were used with first, second and third key respectively. Obtained cipher text has been transformed into their equivalent ASCII codes. For encryption, left shift over the binary form of ASCII was done while for decryption, right shift is done. So, by applying multiple operations, the cipher increases the security. It is a simple and easy to apply technique. Moreover, any technique can be analysed on the basis of number of LOC used, time taken and the space utilized by it. So, analysis had been done on the basis of LOC and time taken by the technique.

2. Related Work

In[1,2] author(s) presents a significant review of various types of vulnerabilities, Structured Query Language Injection attacks, Cross Site Scripting Attack, and prevention techniques. In[3] author(s) discuss and analyse development in field of online authentication process including OTP systems, biometrics and PSTN Network for cardholder authentication. In[4] author(s) exhibits the exploitation of web vulnerabilities in a credit card validation web application using brute force and dictionary attack. In[5] author(s) shows the comparative performance analysis of MD5, DES and AES encryption algorithms on the basis of execution time, LOC (Lines of Code) on web based application(s). In[6] authors also propose a similar technique to handle the security of the alphabets and numbers but without any detailed comparison. In[7] authors propose a technique to encrypt and decrypt the Alphabets, Numbers and Alphanumeric data in minimum span of time with minimum lines of code. In[8] authors have designed a Java based tool to show the exploitation of Injection using SQL Injection attack and Broken Authentication using Brute Force Attack and Dictionary Attack and the prevention of all these attack by storing the data in database in encrypted form using AES algorithm. In[9] authors have explored and exhibited system vulnerabilities and network attacks such as Denial of Service, Brute Force and Dictionary Attack etc.

3. Methodology Adopted

3.1. Explanation of Triplicative Cipher technique

In the theoretical implementation of this technique all the characters of the plaintext have been converted into the assumed encoded number set where 'A'=0, 'B'=1, 'C'=2… 'Z'=25. For this encryption technique three keys K_1, K_2 and K_3 were used. Let's take a string DELHI for which the Triplicative encryption technique is explained theoretically with all of the following given steps:

- Plaintext- DELHI
- Let key K_1=5, K_2=3, K_3=7
- Cipher text C_1= (P*K_1) mod26, C_2= (C_1+K_2) mod26, C= (C_2-K_3) mod26.

Table 1. Key Encryption Process and Shift Operation.

Character	C_1	C_2	C	ASCII	Binary	Left Shift(S_i)
D(3)	(3*5) mod 26 = 15	(21+3) mod26 = 18	(18-7) mod26 =11(L)	76	01001100	01001100(no shift)
E(4)	(4*5) mod 26 = 20	(20+3) mod26 = 23	(23-7) mod26 = 16(Q)	81	01010001	10100010 (1-bit)
L(11)	(11*5) mod 26 = 3	(3+3) mod26 = 6	(6-7) mod26 = 25(Z)	90	01011010	101101000(2-bits)
H(7)	(7*5) mod 26 = 9	(9+3) mod26 = 12	(12-7) mod26 = 5(F)	70	01000110	1000110000(3-bits)
I(8)	(8*5) mod 26 = 14	(14+3) mod26 = 17	(17-7) mod26 = 10(K)	75	01001011	10010110000(4-bits)

- Convert binary to its equivalent decimal and then convert the decimal value into its assigned ASCII character:
  $S = S_0 + S_1 + S_2 + S_3 + S_4$
  S = 1001100 10100010 101101000 1000110000 1001011000
  76 162 360 560 1200
  L ¢ Ũ ¥

  Final cipher text - L¢Ũ ¥.

3.2. Decryption Process:

As the triplicative technique is symmetric key technique, hence the keys that were used for encryption process have been used for decryption process. Right shift operation on the Unicode of the cipher text followed by addition, subtraction and multiplication with the inverse modulo of the key was performed to obtain the plaintext.
Mathematically, the process is carried out as follows:

- Cipher text, C = L¢Ũ ¥.
- Keys, K_1=5, K_2=3, K_3=7.
- Inverse Modulo 5 i.e., K_1^{-1} = 21.

Table.2. Shift Operation and Decryption Process.

Cipher Text	Right Shift	ASCII	Encrypted String	$D_1 = (N + K_3) \bmod 26$	$D_2 = (D_1 - K_2) \bmod 26$	$D = (D_2 * K_1^{-1}) \bmod 26$
L(76)	01001100(no shift)	76	L(11)	(11+7) mod26 = 18	(18-3) mod26 = 15	(15*21)mod26 = 3(D)
¢(162)	1010001 (1-bit)	81	Q(16)	(16+7) mod26 = 23	(23-3) mod26 = 20	(20*21)mod26 = 4(E)
Ũ(360)	1011010(2-bits)	90	Z(25)	(25+7) mod26 = 6	(6-3) mod26 = 3	(3*21)mod26 = 11(L)
(560)	1000110(3-bits)	70	F(5)	(5+7) mod26 = 12	(12-3) mod26 = 9	(9*21) mod26 = 7(H)
¥(1200)	1001011(4-bits)	75	K(10)	(10+7) mod26 = 17	(17-3) mod26 =14	(14*21)mod26=8(I)

Final Plaintext- DELHI.
Mathematically, Plaintext, $P = (D_2 * K_1^{-1}) \bmod 26$
where, $D_2 = (D_1 - K_2) \bmod 26$
where, $D_1 = (N + K_3) \bmod 26$
K_1^{-1} is modulo inverse of the key K_1 and K_2, K_3 are the other two keys.

3.3. Mathematical Modelling

- The characters have been into the assumed encoded number set where, 'A'=0, 'B'=1, 'C'=2,..., 'Z'=25. Let the cipher encryption function be *C(x).* Finally, the Triplicative cipher encryption function $T_n(x)$ has been obtained after applying the necessary mathematical computations. For encryption process following equations resulted in the cipher text :-
  $C(x) = ((C_2(x) - k_3(x)) \bmod 26$
  where, $C_2(x) = (C_1(x) + k_2(x)) \bmod 26$
  and $C_1(x) = ((p(x) * k_1(x)) \bmod 26$
  where the $p(x)$ is the length of plaintext = N_n

  The length for cipher text $C(x)$ w.r.t plaintext $p(x) = N_n$*
  $k(x)$ is the key function used for this cipher technique, where $k_1(x)$, $k_2(x)$ and $k_3(x)$ are the three key functions applied to each letters of plaintext of length N_n .

- After these operations has been applied, the result obtained is a number which must be translated back into a letter, where n belongs to natural numbers i.e.,(1, 2, 3 …….. n). The numerical values for function $C(x)$ is obtained as $C_0(x)$, $C_1(x)$, $C_2(x)$, $C_3(x)$… $C_n(x)$ for length n and the range for $C(x)$ can be defined as $0 < C(x) < 25$. Then translate these numbers into capital letters and use their ASCII code values a_i as (a_0, a_1, a_2, a_3…….a_n) which are in decimal form only.
- Decimal to binary conversion method has been applied for all a_i values as follows:-

  $a_i / 2 = q_0$ (note the value of remainder r_0)

  $q_0 / 2 = q_1$ (note the value of remainder r_1)

  $q_1 / 2 = q_2$ (note the value of remainder r_2) and so on.....$R_n(r_0, r_1, r_2\ r_3\ r_4\ r_5\ r_6\ r_7)$ until the quotient become zero, where $n \in I$ *i.e.*, $R_1\ R_2\ R_3$…….R_n. i.e., up to the length of plaintext.

- Values for each letter has been translated into binary as a function $R_n(x)$ followed by the left shift operation as given as below:

  $S_n = \sum R_n(x + k_i)$ where k_i varies from 0 to n (up to length of plaintext)

  Hence, shift values($s_0\ s_1\ s_2\ s_3$…….. s_n) for each letter has been obtained i.e., 1st letter has no variation.2nd letter has one bit variation and 3rd has two bit variation and so on…and the s_n obtained is of 8 bit length having values like $e_0\ e_1\ e_2\ e_3\ e_4\ e_5\ e_6\ e_7$.

- Binary to decimal conversion method has been applied on each of the shifted values that can be written as:-

  $D_n = e_0 \times 2^0 + e_1 \times 2^2 + \cdots + e_k \times 2^k$

  $= e_0 + (e_1 \times 2) + (e_2 \times 4) + \cdots + (e_k \times 2^k)$ where (n € I) up to length of plaintext.

  $= e_0 + (2 \times e_1) + (2 \times (e_2 \times 2)) + \cdots + (2 \times (e_k \times 2^{k-1}))$

  $= e_0 + 2(e_1 + (e_2 \times 2) + \cdots + (e_k \times 2^{k-1}))$.

- Finally, the numerical values have been obtained for each value of D_n in decimal form and those values have been translated into their ASCII letters. Thus the encrypted cipher text function $T_n(x)$ up to the length of the plaintext i.e., n, is obtained.

- Time calculations for Triplicative cipher function $T_n(x)$:

Table. 3. Time Calculation.

Operation	Time
$C_1(x)$	Δt_0
$C_2(x)$	Δt_1
$C(x)$	Δt_2
ASCII code conversion	Δt_3
Decimal to binary conversion	Δt_4
Shifting operation	Δt_5
Binary to decimal conversion	Δt_6
Conversion from number to ASCII	Δt_7

- Average time taken for the cipher technique:

$T = (\Delta t_0 + \Delta t_1 + \Delta t_2 + \Delta t_3 + \Delta t_4 + \Delta t_5 + \Delta t_6 + \Delta t_7)/8.$

- Time complexity of the system: O(n), where, n is the number of characters of input plain text.

4. Results Obtained

The images below represents the results that were obtained:

Output results obtained after execution of the source code on the given simulation environment (Fig. 2).

Comparison of the technique for three different types of data input source code in the year 2015 (Fig. 3 and Fig. 4).

Table 4. Simulation Environment.

Operating System	Windows 10 Pro 64 bit
Processor	Intel Core i5 3230M
Memory	4 GB
IDE	NetBeans
IDE Version	8.0.2
Language Used	Java
Java Version	1.8.0_25

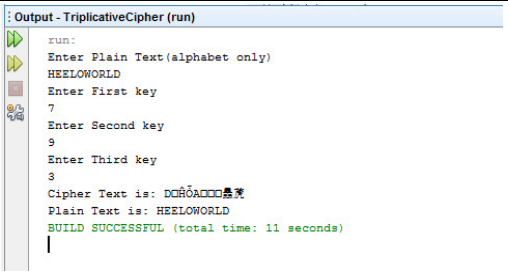

Fig. 1. Output for alphanumeric data.

Similarly the results have been obtained for the numerical and alphabetical input type after executing their respective source codes.

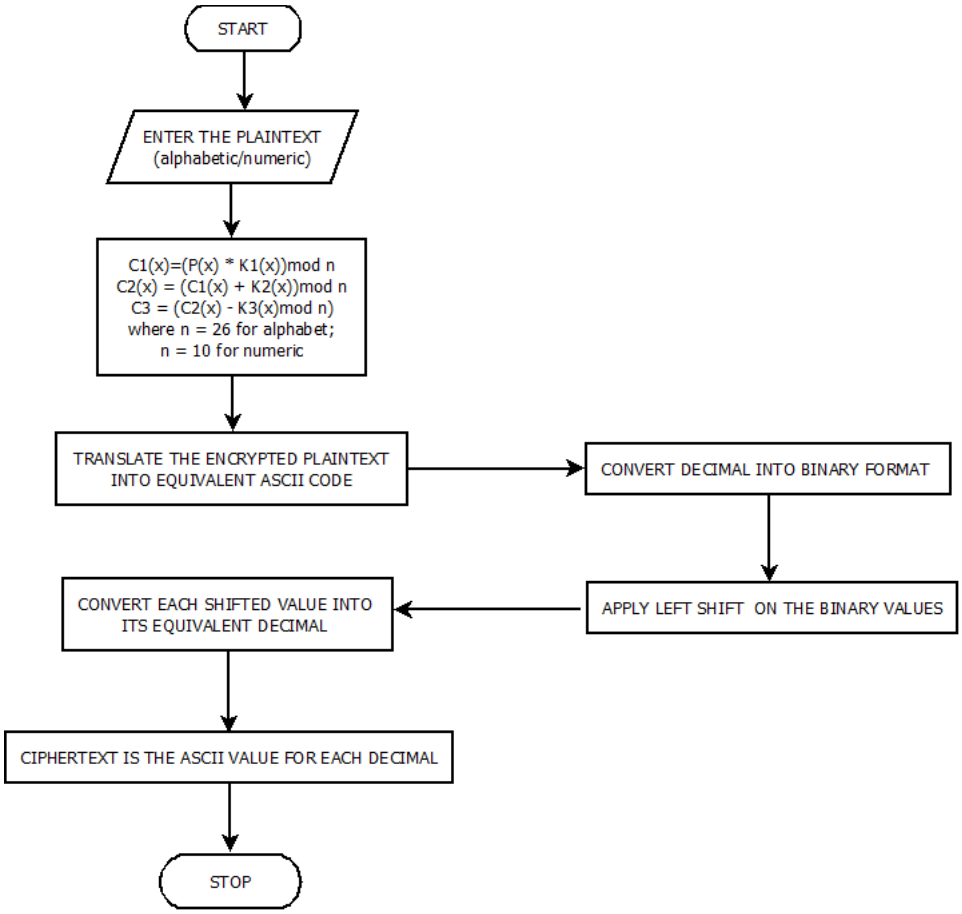

Fig. 2. Flow Chart of the proposed system.

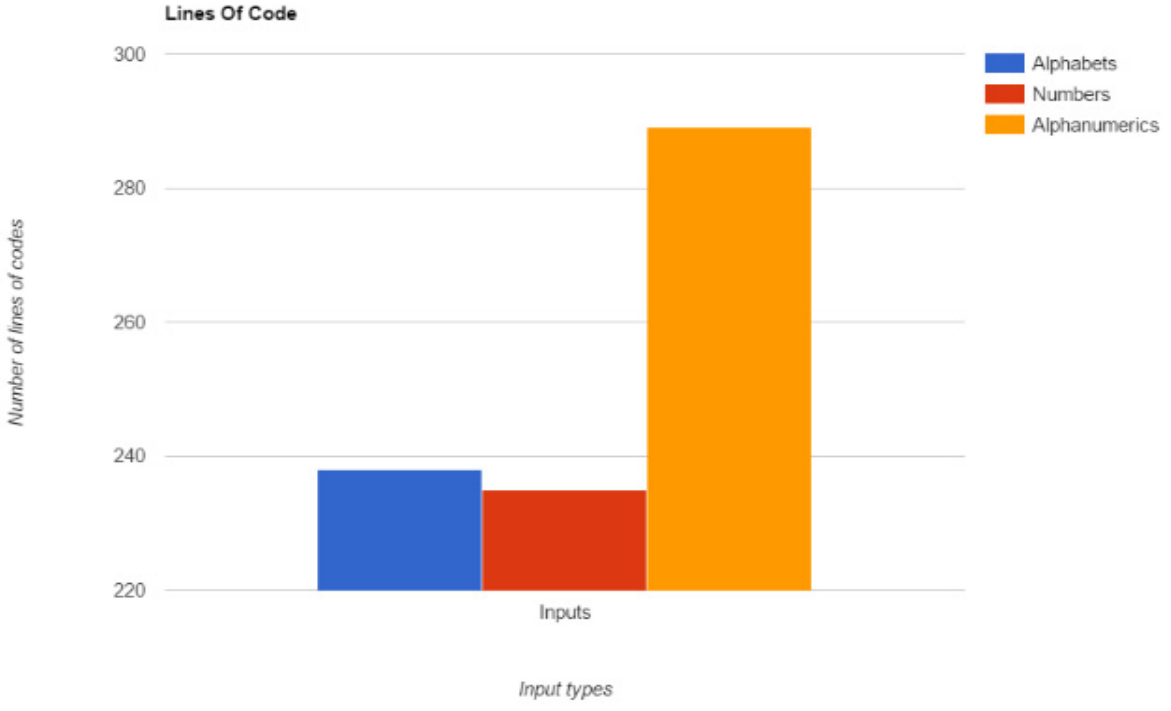

Fig. 3. Comparison of the Running Time. [10]

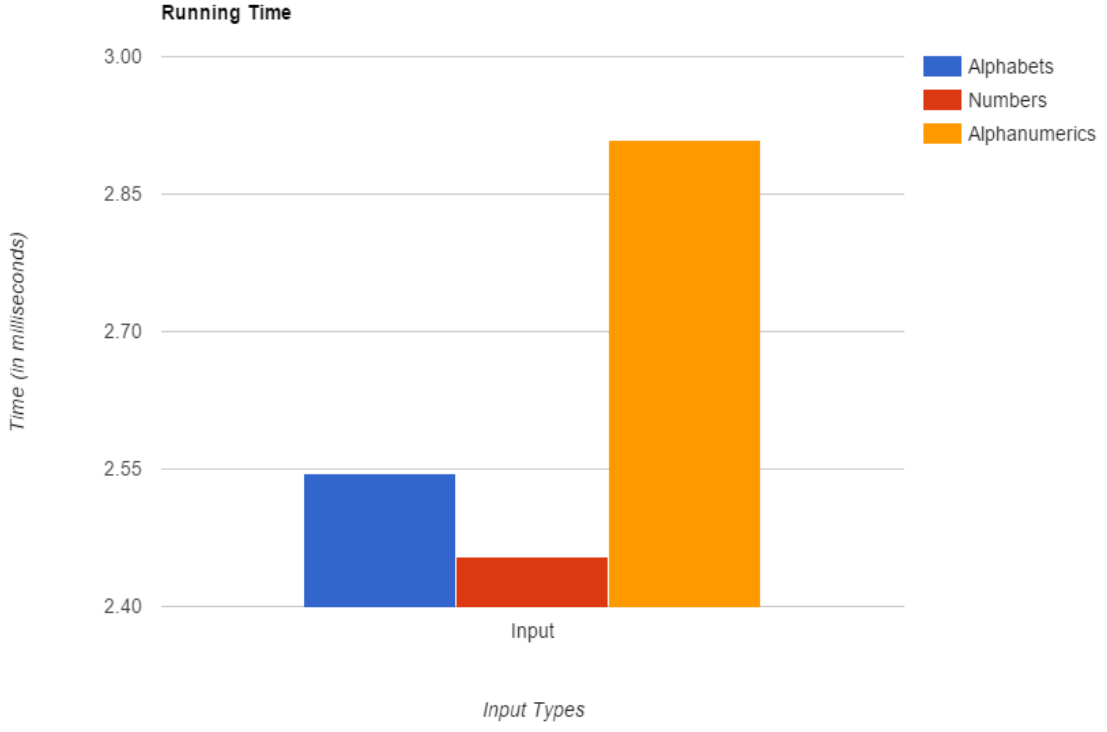

Fig. 4. Comparison of the Line Of Code. [10]

5. Conclusion and Future Work

Gathering from the above data we can safely conclude that the technique gives best results when used on numeric data with the least running time (2.4545 milliseconds) and the least number of lines of code (230 LOC). This paper describes and implements a Triplicative encryption scheme that is robust and searchable encryption technique for data security in computer networks by considering alphabetical, numerical and alphanumerical data. Key generation procedure and implementation for UNICODE using the given technique are some points which will be covered in the near future.

5. References

1. R. Johari and N. Gupta. Secure query processing in delay tolerant network using java cryptography architecture. In Computational Intelligence and Communication Networks (CICN), 2011 International Conference on, pp. 653-657, IEEE (2011).

2. R. Johari and P. Sharma. A survey on web application vulnerabilities (SQLIA, XSS) exploitation and security engine for SQL injection. In: Communication Systems and Network Technologies (CSNT), 2012 International Conference on, pp. 453-458, IEEE (2012).

3. S. Gupta and R. Johari. A New Framework for Credit Card Transactions involving Mutual Authentication between Cardholder and Merchant. In: Communication Systems and Network Technologies (CSNT), 2011 International Conference on, pp. 22-26, IEEE (2011).

4. I. Jain, R. Johari and R.L.Ujjwal. Web Vulnerability Exploitation using Brute Force Attack and Dictionary Attack. In: proceedings of 9th National Conference on Smarter Approaches in Computing Technologies and Applications (SACTA-2014), (2014).

5. R. Johari, I. Jain and R.L.Ujjwal "Performance Analysis of MD5, DES and AES Encryption Algorithms for Credit Card Application" In: International Conference on Modeling and computing (ICMC – 2014), 2014.

6. L. Ruby and Rahul Johari, "Designing a Secure Encryption Technique for Web Based Application", International Journal of Advance Research In Science And Engineering (IJARSE) [ISSN-2319-8354], Volume 3, Issue 7, 159 -163, (July 2014).
7. L. Ruby, Rahul Johari, "SANE: Secure Encryption Technique for Alphanumeric Data Over Web Based Applications", International Journal of Engineering Research and Technology (IJERT) [ISSN NO: 2278- 0181] Volume 3, Issue 8, pp 8-11(August 2014).

8. I. Jain, Rahul Johari, and R.L.Ujjwal "CAVEAT: Credit Card Vulnerability Exhibition and Authentication Tool". In: Second International Symposium on Security in Computing and Communications (SSCC'14), pp 391-399, Springer (2014).

9. Sachin Ahuja, Rahul Johari, and Chetna Khokhar "EAST: Exploitation of Attacks and System Threats in Network". In: Information Systems Design and Intelligent Applications, Advances in Intelligent Systems and Computing (ASIC) Series Volume 339, pp 601-611, Springer (2015).

10. http://www.rapidtables.com/tools/bar-graph.htm.