I J C T A, 9(11) 2016, pp. 5391-5400

EXTENDED CAESAR CIPHER FOR LOW POWERED DEVICES

Priya Verma [*], and Gurjot Singh Gaba[**]

Abstract: Security plays a major role in information exchange through wireless medium as data present in the air interface can be accessed by the unauthorized third party. In today's world, banking sector, companies, etc. usually route their precious information through the internet, which is considered to be the unsecure medium. It has been surveyed that various encryption techniques are built to enhance the security of the user's information. Caesar cipher is one of the most ancient and simple cipher encryption technique. Traditional Caesar cipher is susceptible to brute force attack due to its simplicity in operation. In this paper, a new algorithm is proposed i.e. Extended Caesar Cipher (ECC); to increase the strength of Caesar cipher. ECC and peer techniques are tested on various benchmarks whose results revealed the superiority of ECC over existing ones with huge margins.

Key Words: cryptography; caesar cipher; encryption ; delta formation; security; decryption.

1. INTRODUCTION

CRYPTOGRAPHY comes from Greek work '*Secret writing*'. Cryptography was discovered for safely transmission of knowledge, within the existence of attacker. In cryptography, encryption of information is done at the transmitter end so that no attacker will have access of it while data is on the way. It provides Confidentiality, Integrity, and Accuracy [1]. Cryptography is necessary in data communication or in any network where information is routing from one end to another. Some crucial security needs are:-

- For proving one's identity, authentication should be there.
- Privacy is necessary to make sure that nobody will access the information excluding the destination receiver.
- To make sure the receiver that the information he received has not been changed from the originating point, Integrity should be there.
- To prove that sender actually delivers that message (Non-repudiation).

Essential elements of Cryptography are Plaintext, Ciphertext, Secret key, Encryption algorithm and Decryption algorithm. Cryptography also plays a very important role in user authentication rather than only protecting message from an attacker. In general, two forms of cryptography are discussed: Symmetric key cryptography, and Asymmetric key cryptography; in every case plaintext is converted into some ciphertext so that it will be safely transmitted over the network. Secret key

[*] Discipline of Electronics and Communication Lovely Professional University Phagwara, Punjab, India – 144411
Email: priyaverma1740@gmail.com
[**] Discipline of Electronics and Communication Lovely Professional University Phagwara, Punjab, India - 144411
Email: er.gurjotgaba@gmail.com

cryptography is also known as Symmetric key cryptography. In Symmetric key cryptography, an identical key is used by both the parties involved. The same key is used to decrypt the message which is used by the sender for encryption. Symmetric key cryptography algorithms are well known, DES (Data Encryption Standard), Blowfish, and Advanced Encryption Standard. Symmetric algorithms are popular because their speed enables them to efficiently encrypt large quantities of plaintext [2].

Asymmetric key cryptography works on two dissimilar keys i.e. *Private key* and P*ublic key*. Receiver never shares the private key with anyone but advertises its Public Key. Source uses the general Public Key to cipher the message and recipient can decrypt the message using Private Key. Asymmetric key cryptography algorithms are popularized in the form of RSA algorithm and Diffie Hellman Key Exchange algorithm [3]. Plenty of encryption algorithms persist in the market for secure exchanging of information like Caesar cipher, Playfair cipher, Hill cipher etc. Among all these, Caesar cipher is the most ancient and simplest cipher proposed by Julius Caesar in 20th Century. This notable coding technique of cryptography is used to encrypt the message which we want to transmit over the network. In this technique, every character of the plaintext is interchanged by another character of the language. It is also a kind of substitution cipher, for instance, D would get replaced by A with a left shift of three and so on [4]. In Caesar cipher, only 26 keys are available to encrypt the data, which points out that Caesar cipher is more prone to brute force attack. In this paper, an extended approach of Caesar cipher is proposed, which makes the existing Caesar cipher more strong and powerful so that it bypass potential attacks.

2. TRADITIONAL CAESAR CIPHER

The Caesar cipher is known after its inventor name, Julius Caesar, who in association with Suetonius, helped to shield messages of military for security purpose. Every character of the plaintext is interchanged by a character of some mounted variety of locations down the alphabet. It is also a kind of substitution cipher. In cryptography, a Caesar cipher, is also familiar by the name "shift cipher". This notable coding technique of cryptography is used to encrypt the message which we want to transmit over the network. For instance, with a move of 4, A would be replaced by E, B would be replaced by F and so on [5].

Plaintext	**ABCDEFGHIJKLMNOPQRSTUVWXYZ**
Cipher text	**EFGHIJKLMNOPQRSTUVWXYZABCD**

The formula for ciphertext in Caesar Cipher may be indicated as:

$$c = E(k, p) = (p + k) mod 26 \quad (1)$$

Where E ( ) is the encryption function, 'p' is the plaintext and 'k' is the key.

The formula for decryption is:

$$\mathrm{p} = \mathrm{D}(\mathrm{k}, \mathrm{c}) = (\mathrm{c} - \mathrm{k}) \mathrm{mod} 26 \quad (2)$$

Where D ( ) stands for decryption function, 'c' is the ciphertext and 'k' is the same key used for encryption.

3. RELATED WORK

Lim Chong Han (2014), et al. in paper entitled "*An Implementation of Caesar Cipher and XOR Encryption Technique in a Secure Wireless Communication*" proposed a technique for the secure transmission of data over the networks, which involves three stages i.e. Encryption design technique, serial port communication program and Encoding pattern design. The secret writing

technique utilized in this paper is Caesar cipher. Caesar cipher is made complicated by merging existing *Caesar cipher technique* with *XOR* secret writing. The *XOR* secret writing uses bitwise *XOR* operations to obtain the required complexity and randomness in Ciphetext. A comparison on the basis of Knowledge Co-ordinated Universal time has been done to prove the efficiency of suggested technique over the existing ones. The mixing of *Caesar cipher* and *XOR* secret writing has not affected the interval of knowledge and has evidenced to supply a secure wireless communication [6].

A. Rajan (2014), et al. in paper entitled "*Advancement in Caesar cipher by randomization and delta formation*" states that by using relative frequency analysis, Caesar cipher can be deciphered through computer easily. In this paper, they did improvement in Caesar cipher using principle of randomization and delta formation methodology. Here the secret writing method is splitted into 3 components i.e. Randomization, Encryption and Delta formation. In this methodology, key table is generated indiscriminately; it is terribly tough for the assailant to guess the key table. Neither is it cracked by cryptanalyst nor by brute force analysis. This methodology conjointly cannot be cracked by frequency analysis. This methodology can generate potential combinations of 26!*9! i.e. Undecillion Combinations. Here, the life time of the projected system: $(1.4346 \times 10^{32}) / (348 \times 10^{8}) = 4.20 \times 10^{21}$ seconds, that is around 1.334×10^{13} years. Hence this method is categorically secure [7].

K. Goyal (2013), et al. in paper entitled, "*Modified Caesar Cipher for Better Security Enhancement*" portrayed that the existing model of Caesar cipher uses the same key for encryption which cannot prevent the system from attack in harsh environment. So they proposed a methodology, where the value of character is incremented by one if the alphabet index of the plaintext is even and decrement the value of letter by one if the index of the plaintext is odd. Security provided by this methodology will be increased more by applying additional algorithms over data [8].

Shahid Bashir Dar (2014), et al. in paper entitled "*Enhancing the Security of Caesar Cipher Using Double Substitution Method*" demonstrates that adding complexity in functioning of the algorithm can strengthen its impact to withstand against severe attacks. Double substitution is performed in order to make Caesar cipher more secure and stronger so that it can be protected from cryptanalyst and brute force attack. Initially, the plaintext is reshaped in the form of matrix. Order of the columns is determined by the key 'K_1'. By reading the message column by column, we get the ciphertext CT_1. Then shift each character of the CT_1 by using key value k_2 which gives the final ciphertext CT_2. The decryption is carried out in the reverse manner. This method uses very less structured permutation and claims to overcome the limitations of simple Caesar cipher [9].

4. PROPOSED TECHNIQUE (ECC)

In existing Caesar cipher, every character of the plaintext is interchanged by a character with some other character in the language. It has only 26 possible set of keys to encipher the data. So by trying all the possible set of keys, brute force attacker can easily attack on it. To overcome the limitations of already existing Caesar ciphers, a new technique is proposed in this paper which is named as *Extended Caesar Cipher (ECC)*.

Methodology: Proposed technique comprises of three parts:-

(a) Key generation process

(b) Encryption process

(c) Decryption process

(a) ***Key Generation Process:*** To make our plaintext more secure, we need a strong key while exchanging the data over the network. In the encryption process, firstly the plaintext is converted into binary form i.e. 8 bit value. Take any value of key and calculates the factorial of it and then multiply it by two. Now convert the key value into its binary form. To make the key value equivalent to 8 bits, add the compliment of key value to the MSB and then combines both to get the final key value.

Plaintext	*pic*
Key	3

According to the algorithm, convert every character of plaintext into its ASCII binary equivalent i.e. for 'p', the binary form is '01110000', binary form of 'i' is '01101001'and for 'c' is '01100011'.

Factorial of Key (K) = 3! = 6

Multiplication; K × 2 = 6 × 2 = 12

Binary equivalent of K = 1100.

Compliment; K1=0011

Affix K1 to MSB of K to get the final key (K2) i.e. 00111100 which is used for the encryption process.

(b) ***Encryption Process:*** In the second part, complex encryption technique is designed to encrypt our message (plaintext) so that no attacker will attack on it and it will be safely transmitted over the network. For encryption, let us take the first character of plaintext '*pic*' i.e. 'p'.

Plaintext	*pic*
Binary equivalent of plaintext 'p'	01110000
Key value (K_2)	00111100
First Cipher text result (C_1)	10110011
'NOT' operation on C_1	01001100
Second Cipher text result (C_2)	01001100

Now XOR the last 4 bits of key (K_2) with the first 4 bits of the plaintext character '*p*' and the first 4 bits of key (K_2) with the last 4 bits of plaintext. At the last, shuffle the bits of C_2 according to the predefined pattern cited in the table. The bits are shuffled to enhance randomness amongst the bit pattern.

Bit Location	1	2	3	4	5	6	7	8
Shuffle Pattern	1	8	6	2	3	4	7	5

Final ciphertext obtained after shuffling (C_3) = 00110001.

Decimal value of C_3 is '49'and the char value is '1'. So 'p' is encrypted as '1'. Similar process is used to obtain the encrypted values for 'i' and 'c'. Therefore encryption of '*pic*' resulted in Ciphertext '1tw'.

(c) ***Decryption Process:*** The decryption part is exactly the inverse of the encryption part.

5. RESULTS AND DISCUSSION

Result helps to evaluate the effectiveness of the system. For testing the efficiency of the ECC, various tests are conducted while taking the plenty of inputs into consideration.

5.1 Avalanche Effect

A small change in either the plaintext or the key, should produce a significant change in the ciphertext. This is called as Avalanche effect. So it is expressed mathematically as, the ratio of number of flipped bits in the ciphertext to the total number of bits in the ciphertext [10].

$$\text{Avalanche effect} = \frac{\text{No. of flipped bits in the ciphertext}}{\text{No. of bits in the ciphertext}} * 100 \quad (3)$$

Results obtained after calculating the Avalanche effect of various techniques is highlighted in Table1.

Table 1
Avalanche Effect Analysis

Encryption techniques	*Key*	*% age of bits flipped for different keywords*			
		Bad	*Back*	*Guilt*	*Beauty*
Basic Caesar (B.C) [5]	3	29.16	28.12	30	31.25
XOR technique [6]	3	25	12.50	32.50	27.08
Delta formation [7]	3	NA	NA	NA	NA
Extended Caesar Cipher (ECC)	3	41.66	37.50	40	33.33

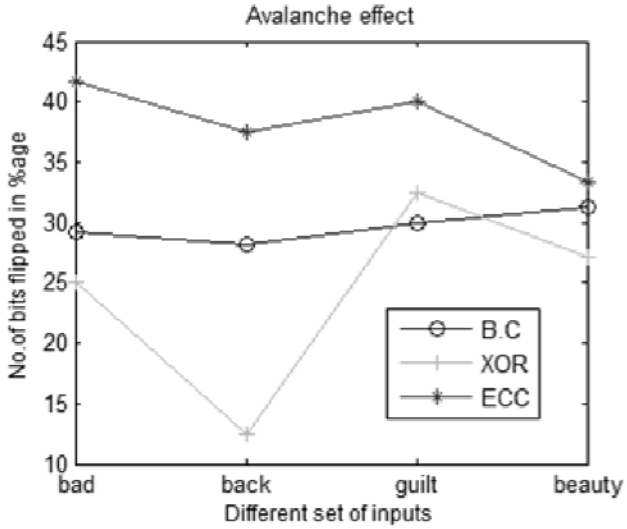

Figure 1. Comparison of different algorithms based on Avalanche Effects.

Fig.1 shows that proposed technique (ECC) results with a high percentage of Avalanche effect i.e. more no. of bits are flipped as compared to existing algorithms i.e. Basic Caesar (B.C), and Caesar using XOR technique. This effect is not visualized for delta formation and randomization technique, as the length of the ciphertext and plaintext is unequal. Delta formation technique not only adds redundancy but also consumes more time and processing power. The test proved the perseverance of randomness in ECC for various inputs.

5.2 Run Test

It is done to check the randomness of the binary sequence. A run is an uninterrupted sequence of identical bits. More number of runs indicates better randomness. In particular, this test determines whether the oscillation between such zeros and ones is too fast or too slow.

$$P_{value} = \text{erfc}\left(\frac{V_{n(obs)} - 2n\pi(1-\pi)}{2\sqrt{2n\pi}(1-\pi)}\right) \quad (4)$$

Where $V_{n(obs)}$= the total no. of runs across all n bits. If the p-value is greater than 0.01 then conclude that sequence as random otherwise it is non random.

Table 2
Run Test Analysis

Encryption Techniques	*Key*	*No. of Runs for different keywords (%age)*			
		See	*Blow*	*Hello*	*Flower*
Basic Caesar (B.C) [5]	3	0.54	0.59	0.57	0.58
XOR technique [6]	3	0.45	0.43	0.52	0.56
Delta formation [7]	3	0.54	0.38	0.31	0.25
Extended Caesar Cipher (ECC)	3	**0.83**	**0.62**	**0.67**	**0.66**

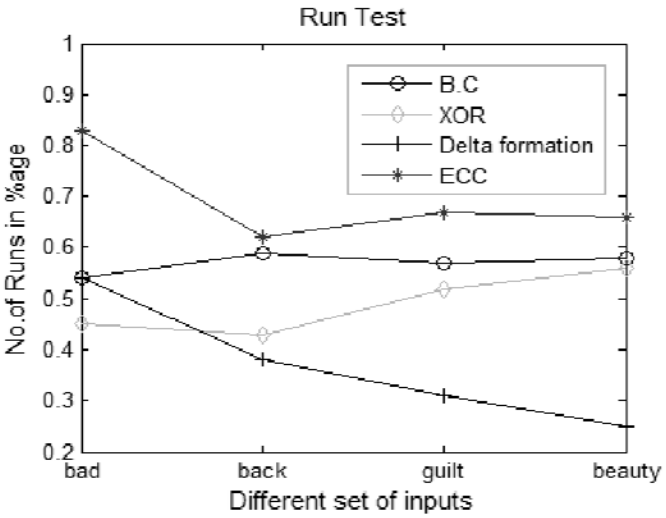

Figure 2. Comparison of different algorithms based on RUN test.

Table 2 shows that a run test is conducted on all the algorithms. It has been checked by taking different inputs to find the runs of the sequence. More the number of runs better will be the technique. Figure 2 shows that our proposed technique (ECC) results in more number of runs in the sequence than all existing algorithms which makes it difficult for an attacker to predict the plaintext.

5.3 Frequency Test

The motive of this test is to check whether the number of ones and zeroes in the sequence are equal or unequal. Equivalent proportion of zeroes and one is expected for the truly random sequence [10]. Firstly the zeroes and ones of the input sequence is converted to values of -1 and +1 and added together to produce S_n , then Compute the test statistic (*sobs)* and p-value. If the p-value is > than 0.01, then conclude that sequence is random otherwise it is non random.

Where $$sobs = \frac{|Sn|}{\sqrt{n}} \quad (5)$$

$$P_{value} = erfc\frac{sobs}{\sqrt{2}} \quad (6)$$

where *erfc* is the complimentary error function.

Table 3
Frequency Test Analysis

Encryption techniques	*Key*	*Frequency of 0's and 1's for different keywords (%age)*			
		Bad	*aays*	*abcde*	*abcdef*
Basic Caesar [5]	3	0.5000	0.5000	0.4750	0.4792
XOR technique [6]	3	0.4583	0.5313	0.4250	0.4375
Delta formation [7]	3	0.5288	0.5577	0.5577	0.5577
Extended Caesar Cipher (ECC)	3	**0.5417**	**0.5625**	**0.5750**	**0.5625**

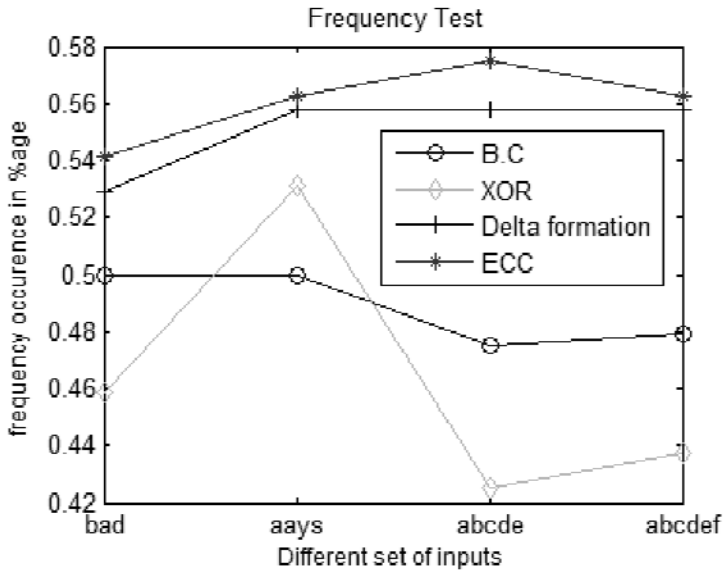

Figure 3: Comparison of different algorithms based on Frequency test

Table 3 shows the response of different techniques in the frequency test for multiple inputs. We can clearly see from the fig. 3 that our proposed technique produced high degree of the randomness in the binary sequence as compared to all the existing algorithms.

5.4 Binary Derivative Test

The binary derivative test is used to measure the randomness of the binary sequence. The first binary derivative of S_1, $D_1(S_1)$, is the binary string of length n - 1 formed by XORing adjacent pairs of digits.

$$P_{avg} = \frac{S_1}{n} \quad (7)$$

If the value of P_{avg} is near to 0.5 than the sequence is random otherwise it is non random.

The following results are obtained after calculating the respective Binary derivative test.

Table 4
Binary Derivative Test Analysis

Encryption techniques	*Key*	*Binary Derivative test for different keywords (%age)*			
		abc	*abcd*	*quick*	*pickle*
Basic Caesar (B.C) [5]	3	0.4614	0.3286	0.4695	0.4866
XOR technique [6]	3	0.2000	0.3383	0.4151	0.2795
Delta formation [7]	3	0.4495	0.4495	0.4645	0.4489
Extended Caesar Cipher (ECC)	3	0.4752	0.4530	0.4938	0.4908

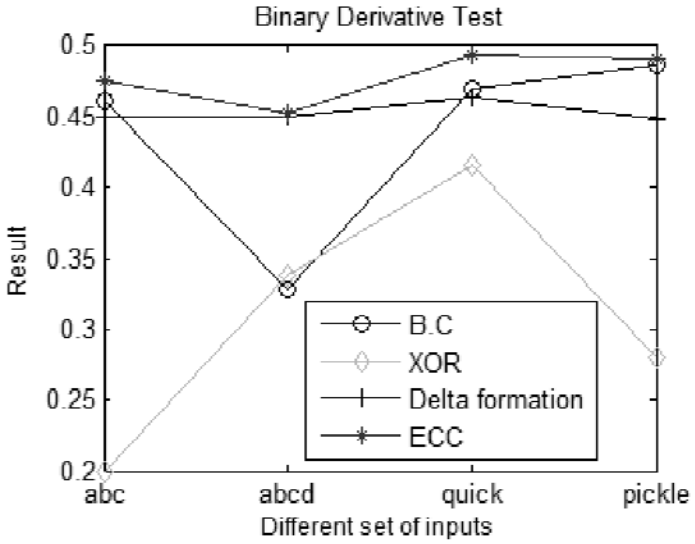

Figure 4. Comparison of different algorithms based on Binary Derivative test.

Table 4 shows the different inputs used to calculate the binary derivative test; in order to verify that randomness of the binary sequence. We can see clearly from the results in figure 4 that our proposed technique (ECC) surpassed the benchmark of accuracy. ECC is found to be more suitable in networks as it has marked its significance over existing techniques.

6. CONCLUSION

Caesar cipher is a very simple technique used for encryption in cryptography, but it is insecure due to less number of keys, and it is more prone to attacks. To surpass the limitations of the existing Caesar cipher algorithms, a new technique is designed in this paper i.e. Extended Caesar Cipher (ECC). From the discussions, we can clearly see that our proposed technique (ECC) has higher Avalanche effect, more runs in Run test, more equalization in Frequency test and desired response in Binary derivative test; which shows that our proposed technique put stronger impact on encryption result than all other existing algorithms.

References

[1] R. Mane "A Review on Cryptography Algorithms, Attacks and Encryption Tools," *International Journal of Innovative Research in Computer and Communication Engineering,* vol. 3, no. 9, pp.8509-8514, 2015.

[2] G. Gupta, R. Chawla, "Review on Encryption Ciphers of Cryptography in Network Security," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 2, no. 7, pp.1-26, 2012.

[3] S.Chandra, "A comparative survey of Symmetric and Asymmetric Key Cryptography," *International conference on Electronics, Communication and Computational Engineering (ICECCE)*, Hosur, pp. 83–93, 2014.

[4] S. Shakti, "Encryption using different techniques," *International Journal in Multidisciplinary and Academic Research (SSIJMAR)*, vol. 2, no. 1, pp. 1-9, 2013.

[5] O. Abraham, Ganiyu O. Shefiu, "An improved Caesar cipher algorithm," *International Journal of engineering science and advanced technology*, vol. 2, pp.1199-1202, 2012.

[6] Lim Chong Han, N.M. Mahyuddin, "An Implementation of Caesar Cipher and XOR Encryption Technique in a Secure Wireless Communication," *2nd International conference on Electronic Design (ICED)*, Penang, pp.111-116, 2014.

[7] Rajan, D. Balakumaran, "Advancement in Caesar cipher by randomization and delta formation," I*nternational conference on Information communication and Embedded systems (ICICES)*, Chennai, pp.1-4, 2014.

[8] K. Goyal, S. Kinger, "Modified Caesar Cipher for Better Security Enhancement," *International Journal of Computer Applications*, vol. 73, no. 3, pp. 27-31, 2013.

[9] S.B. Dar, "Enhancing the Security of Caesar Cipher Using Double Substitution Method," *International Journal of Computer Science & Engineering Technology*, vol. 5, no. 7, pp.772-774, 2014.

[10] William Stallings, "Cryptography and Network Security: Principles & Practices", New York, NY: Pearson Education, 2006.