Digital Image Encryption using Logistic Chaotic Key-based RC6

Mohammed Baz
College of Engineering and Information Technology
Taif University, Taif, K.S.A.

ABSTRACT
RC6 is a symmetric block cipher that possesses remarkable features, e.g., simple structure, Feistel structure and supporting for different block size, key length and number of rounds. However, some recent studies show that this cipher is subject to several cryptanalyses such as statistical attack, linear cryptanalysis, correlation attack and brute force attack. This paper proposes an enhancement version dubbed Chaotic Key Based RC6 (CKBRC6) that makes use of the Logistic map to generate round keys. Comprehensive assessments for the security of our proposal and fair comparisons between it and RC6 demonstrate the outperformance of the former in the domain of image encryption.

General Terms
Security, Digital image processing, Block cipher.

Keywords
RC6, Logistic map, Chaotic encryption key.

1. INTRODUCTION
The fast pace with which communication and information technologies are developed has prompted unprecedented opportunities to generate, store and exchange massive amount of visual data over public networks. According to [1] during 2016, about 2.2 trillion photos were published to the Internet and its expected that this figure is going to grow exponentially due to the appearance of new visual analytical applications such as visual Internet of Thing [2] and Big visual data [3]. The paramount concern in handling such volume of data is how to maintain their privacy and security, Online Trust Alliance testified that the number of reported attack-incidents has been doubled during 2017 compared to 2016 [4]. This, in turn, raised the need to use encryption schemes to protect data; [5] highlighted that in 2016, 40% of websites traffic were encrypted and by 2019, it is expected that this percentage reaches about 80%.

Although there is an enormous number of ciphers, the appealing features of Rivest cipher (RC6) [6] make it one of the good candidates for image encryption. RC6 is a symmetric block cipher that has been evolved from one of the widely-used cipher RC5[7] with the several enhancements that were made to address the requirements of Advanced Encryption Standard (AES) competition such as: Doubling the number of working registers from 2 to 4 to speed up the execution time; Adopting of multiplication as a primitive operation to increase the diffusion per round; using two rotations per round to strengthen the withstanding to linear and differential attacks and employing of a quadratic function to accelerate approaching the avalanche effect. Besides these enhancements, RC6 is easy to implement and has a versatile structure that enables users to parametrize the length of the encryption key, block size and number of rounds without needing for reconfiguration.

Nonetheless, some recent studies reported that RC6 can be broken under some attacks such as statistical attack, linear cryptanalysis, correlation attack and brute force attack [8-14]. One of the main reason behind the vulnerability of RC6 to attack is the ability to predict it cryptographic keys. Basically, RC6 likes many other block ciphers make use of key schedule routine to extend the secret key provided by the user into a set of keys that are used during encryption and decryption rounds. RC6 employs Linear Congruential Generator (LCG) with the aid of two predefined magic constants to initialize the key rounds and then mixes them with user's key to generate such round keys. The main shortcoming of LCG is that a pseudorandom sequence generated by this algorithm is predictable and can be recovered readily [15-21].

This paper proposes a Chaotic Key Based for RC6 (CKBRC6) as enhancement version for RC6 in which the key schedule routine is replaced with a more powerful routine using the logistic chaotic map [23]. A justification for using a chaotic map in key schedule routine emerges from the fact that a good key schedule must produce unpredictable, uncorrelated and highly randomness rounds keys using a deterministic function that can generate the same sequence at both encryption and decryption routine identically. Thereby, a key can be spread over many digits of ciphertext which in turn conceals statistical structure of plaintext and enrich the resistance of chipper to cryptanalysis. The logistic map has been employed here because it is extremely sensitive to initial conditions. A study presented in [24] pointed out a tiny distributing of order 10^{-30} in the initial condition of the logistic map can produce a completely different series after just 99 generations. This not only hinders the ability of attacks to predict the generated sequence but also widens the spectrum from which the key can be generated. Moreover, the deterministic and simple form of the logistic map imposes no further overhead nor computational complexity on encryption and decryption routines. This is of tremendous importance in image cryptography due to the need for a fast and computational effective cipher that can be executed in real-time. The performance of the proposed cipher CKBRC6 is assessed from different perspectives including visual testing, key space analysis, cipher cycle analysis, correlation coefficient analysis, information Entropy analysis and key sensitive analysis. The results of these assessments exhibit the advantages of CKBRC6 and its surpass compared to RC6.

The body of this paper is organized as follows: section 2 overviews the works related to cryptanalyses of RC6 and the enhancements reported in the open literature. Section 3 provides the proposed cipher and in section 4 the security analysis of CKBRC6 is given. Finally, in section 5 conclusion of this work is provide.

2. RELATED WORK
Owing to the popularity of the RC6 algorithm, several cryptanalyses were conducted on it, this section reviews their

findings and highlights some of the works proposed to improve this cipher.

The χ^2 attack is one of the effective cryptanalysis that has been applied to RC6 under different assumptions; in [9, 10] the authors show that the χ^2 key recovery attack can be derived straightly from the distinguishing attack. Furthermore, they pointed out that the reason behind the success of χ^2 attack is that some rotations in the RC6 yield a correlation between the plaintext and ciphertext. The results of [10] show that χ^2 attack can break RC6 with 128-bit key and 12 rounds with just 2^{94} plain text, with 192-bit key and 14 rounds with 2^{108} and with 256-bit key and 15 rounds with just 2^{119} text. Another work presented in [11] assessed the withstanding of RC6 without pre-or-post whitening (RC6W) to χ^2-variance attack and showed that with $2^{123.9}$ texts, the RC6 with 128bit key can be broken after 17 rounds. [12] applied the same attack as [11] expect that it considered RC6 without post whitening (RC6P) and showed that with $2^{117.84}$ texts, the RC6 with 128bit key can be broken after 16 rounds. The ability of Linear cryptanalysis to break RC6 was investigated in [13] and showed that after 16 rounds with 2^{119} text the RC6 can be broken. Multiple linear attack was also conducted in [14] and shows that with 2119,68 text and 192-bit key, RC6 can be broken at 14 rounds. The work presented in [8] proves theoretically that for both 192-bit and 265-bit key and up to 16 rounds and $2^{127.20}$ RC6 can be broken. Table 1 summarized the findings of these works.

Table I. results of RC6 cryptanalysis

Type of attack	Cipher parameters	Number of rounds	Number of plain text
Linear attack	RC6	16	2^{119}
Multiple linear attack	192-bit key RC6	14	$2^{119.68}$
χ^2 attack	128-bit key RC6	12	2^{94}
	192-bit key RC6	14	2^{108}
	256-bit key RC6	15	2^{119}
χ^2 attack	128-bit key RC6W	17	$2^{123.9}$
χ^2 attack	128-bit key RC6P	16	$2^{117.84}$
χ^2 attack	128-bit key RC6	8	$2^{63.13}$
	192-bit key RC6	16	$2^{127.20}$
	256-bit key RC6	16	$2^{127.20}$
χ^2 attack	128-bit key RC6	12	$2^{109.21}$
	192-bit key RC6	16	$2^{127.57}$
	256-bit key RC6	16	$2^{127.57}$

Motived by these results, several enhancements for RC6 have been proposed. The authors of [25] proposed adding another swiping function in each round in order to strengthen the security of RC6. Their results showed that under χ^2 attack, the proposed cipher with 13 rounds can give the same results of RC6 with 16 rounds. The work proposed in [26] aims to improve the Feistel structure of RC6 by redesigning the encryption and decryption algorithm in such a way that makes them identical. This proposal simply adds symmetry layer between the encryption and decryption algorithms. Security tests presented in this work shows the surpass of the enhancement version compared to RC6. The work presented in [27] investigated the case when RC6 is applied to an image with odd pixel size, it is shown that under this circumstance, the original image cannot be retrieved correctly due to mismatching between the number of pixels and division of the block and the need to padding the block registers. This proposal reduced the block size of original RC6 to 24 bits and modified the values of magic constants accordingly.

3. THE PROPOSED ALGORITHM

This paper introduces a CKBRC6 as an enhancement version of RC6 that can overcome its main shortcomings without intact its appealing features. For the sake of clarity, our proposal follows the structure of most block cipher that divided algorithm into three routines: key schedules, encryption and decryption. Moreover, the proposed algorithm can work with different word size (denoted here by A), number of rounds (denoted by B) and length of block (denoted by C). Hence the short-notation CKBRC6-$A/B/C$ is used to refer to word size, number of rounds and length of block. Moreover, CKBRC6 adopts the primitive operations defined in RC6 that can be summarized as: $\alpha + \beta$, $\alpha - \beta$ and $\alpha \times \beta$ which refer to addition, subtraction and multiplication of two integer numbers α and β modulo 2^A respectively. The $\alpha \lll \beta$ is used to denote to the left rotation of α by the amount given in the least significant $\log_2 A$ bits of β. Similarly, $\alpha \ggg \beta$ represents the right rotation of α by the amount given in the least significant $\log_2 w$ bits of β. Finally, $\alpha \oplus \beta$ is used to signify Binary Exclusive-OR of α and β.

The following subsection overviews the logistic map that is used to generate the round keys and then describes the key schedule, encryption and decryption routines of the CKBRC6 cipher.

3.1 Overview of Logistic Map

The logistic map is a 2nd degree polynomial map that has been widely used to model nonlinear dynamic system due to its ability to represent different chaotic characteristics; this map is given by the following recursive equation:

$$f(x) = x_{n+1} = \lambda x_n (1 - x_n) \qquad (1)$$

where $x_n \in [0,1]$ is the phase space of the map, λ is the system parameter that can take values range [0,4] and n represents the number of iterations. Owing to the ability of the logistic map to generate unpredictable and uncorrected characteristics, it has been used in the field of cryptography as a pseudorandom number generator, cryptographic algorithm and cryptanalysis [28-31].

This study uses the logistic map to design the chaotic key schedule for RC6, hence it adopts those regions at which the complete chaotic characteristics are generated. One of the effective ways to test this region is the Lyapunov exponent [24] which is used to quantify the speed with which two near trajectories are evolved. According to [23-24] the region at which the Lyapunov exponent is always positive is the region that has the complete chaotic behavior. In order to visualize this region figure1 shows the Lyapunov exponent for logistic map, as it can be seen from this figure, the interval [3.999,4] is the best choice, hence we use it here to generate the round keys.

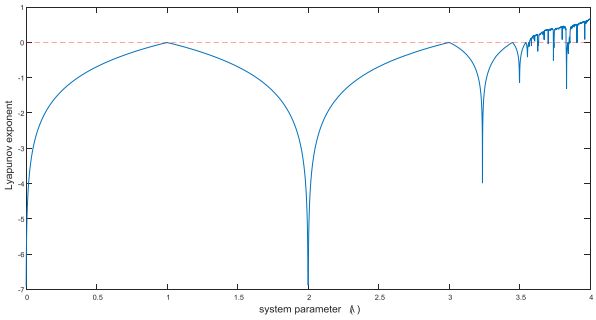


Figure 1 Lyapunov exponent of logistic map

3.2 Key Schedule Routine

The inputs of this routine are the key supplied by the user that is accumulated in vector Usr, the number of rounds B, and a temporary vector dubbed $Temp$. This routine starts by defining a $Temp$ vector and then uses the $logistic$ function to populate this array with the output of the function $logistic$ that takes two arguments the first is the ordered elements of user's key and the second is the number of iterations. Since the first input is of type byte and the input of a logistic map is a number within the range [0.1], the function $\varphi_1(x)$ is defined to convert these bytes to the interval [0,1] and then $\psi(x)$ is used to return an integral part of the outcome. By the end of this section the Temp function is populated with the output of the logistic map and its ready to be mixed with the Usr to enhance the confusion and diffusion operations of the proposed cipher. Here the function $\varphi_2(x)$ is used to convert the output of the $\varphi_1(x)$ into the words, while the c variable is used to keep track of the number of rounds when the length of the user's key is not equal to the length of Temp vector, hence the shorter vector is repeated more which in turn ensures that the outcome of this routine possesses significant amount of one-wayness which hinders determine the round key Q from the user's key.

$Usr \leftarrow key\ provided\ by\ the\ user$

$Temp[0] \leftarrow init\_pad$

$Subkey \leftarrow 0,$

$for\ i = 1\ to\ 2B + 3\ do$

{

$Temp[i] = logistic(\psi(\varphi_1(Usr[subkey]) + pad), Usr[next\_subkey(Subkey)]) + Temp[i - 1]$

}

$for\ i = 1\ to\ 2B + 4\ do$

{

$Q[i] \leftarrow \varphi_2(Temp[i]);$

$i \leftarrow j \leftarrow 0$

$v \leftarrow \max[c, 2B + 4]$

$for\ s = 1\ to\ v\ do$

{

$Temp[j] \leftarrow Temp[j] \lll \log_2 A$

$Q[i] \leftarrow (Q[i] + Temp[j]) \lll < (\log_2 A + Temp[j])$

$i \leftarrow (i + 1)\ mod\ (2B + 4)$

$j \leftarrow (j + 1)\ mod\ c$

}

}

3.3 Encryption Routine

The input of encryption routine are four registers encompasses plains text denoted here by: $\mathcal{X}, \mathcal{Y}, \mathcal{Z},$ and $\mathcal{W}$ each of which is of length A words. In additional to the round keys generated from the key schedule routine, i.e., $Q[0, \ldots, 2B + 3]$ and number of rounds. The main aim of this routine is to covert the plain-text populated in the four registers to the cipher-text.

$\mathcal{Y} \leftarrow \mathcal{Y} + Q[0]$

$\mathcal{W} \leftarrow \mathcal{W} + Q[1]$

$for\ i = 1\ to\ B\ do$

{

$t \leftarrow (\mathcal{Y} \times (2\mathcal{Y} + 1)) \lll \log_2 A$

$u \leftarrow (\mathcal{W} \times (2\mathcal{W} + 1)) \lll \log_2 A$

$\mathcal{X} \leftarrow ((\mathcal{X} \oplus t) \lll u) + Q[2i]$

$\mathcal{Z} \leftarrow ((\mathcal{Z} \oplus u) \lll t) + Q[2i + 1]$

$(\mathcal{X}, \mathcal{Y}, \mathcal{Z}, \mathcal{W}) = (\mathcal{Y}, \mathcal{Z}, \mathcal{W}, \mathcal{X})$

}

$\mathcal{X} \leftarrow \mathcal{X} + Q[2B + 2]$

$\mathcal{Z} \leftarrow \mathcal{Z} + Q[2B + 3]$

3.4 Decryption Routine

In the decryption routine the four registers $\mathcal{X}, \mathcal{Y}, \mathcal{Z},$ and $\mathcal{W}$ contains the cipher-text are fed into this routine along with the number of round (B) and round key $Q[0, \ldots, 2B + 3]$. The output of this routine is the plaintext.

$\mathcal{Z} \leftarrow \mathcal{Z} - Q[2B + 3]$

$\mathcal{X} \leftarrow \mathcal{X} - Q[2B + 2]$

$for\ i = B\ to\ 1\ do$

{

$(\mathcal{X}, \mathcal{Y}, \mathcal{Z}, \mathcal{W}) \leftarrow (\mathcal{W}, \mathcal{X}, \mathcal{Y}, \mathcal{Z})$

$u \leftarrow (\mathcal{W} \times (2\mathcal{W} + 1)) \lll \log_2 A$

$t \leftarrow (\mathcal{Y} \times (2\mathcal{Y} + 1)) \lll \log_2 A$

$\mathcal{Z} \leftarrow ((\mathcal{Z} - Q[2i + 1]) \ggg t) \oplus u$

$\mathcal{X} \leftarrow ((\mathcal{X} - Q[2i]) \ggg u) \oplus t$

}

$\mathcal{W} \leftarrow \mathcal{W} + Q[1]$

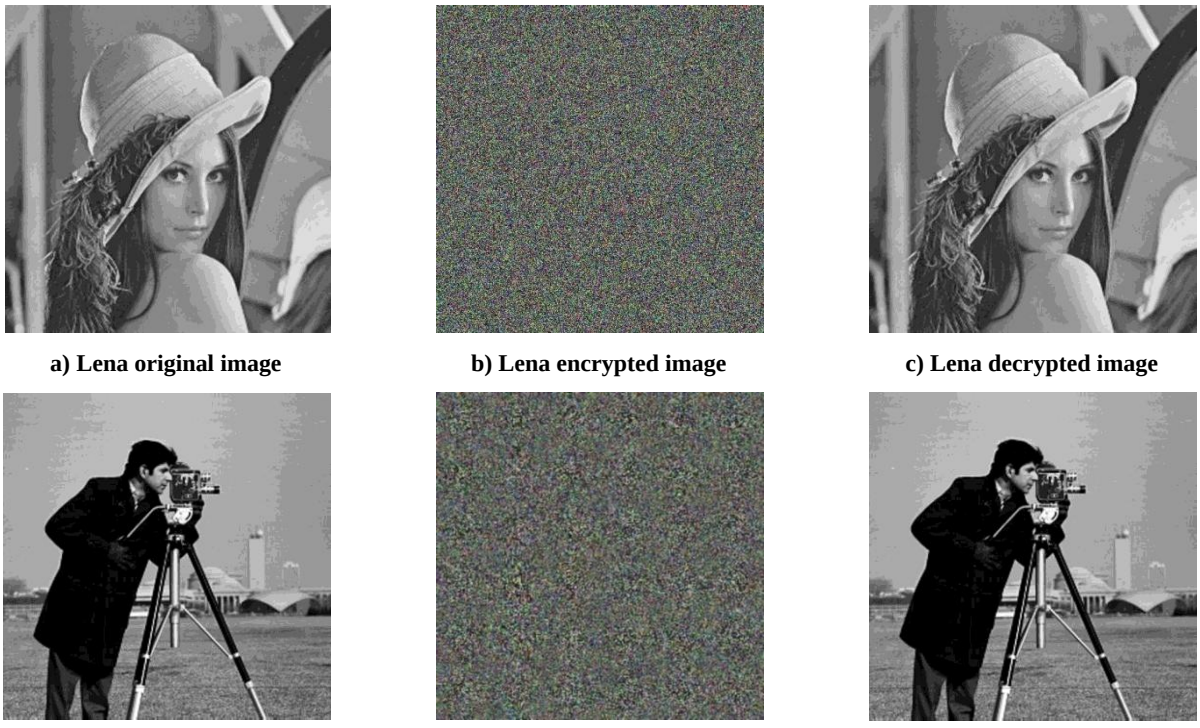
$\mathcal{Y} \leftarrow \mathcal{Y} + Q[0]$

4. SECURITY ANALYSIS OF CKBRC6 AND TEST RESULTS

This section assesses the immunity of CKBRC6 to well-known cryptanalyses such as: statistical, correlation, entropy, brute-force, frequency and differential attacks by applying it on several digital images. In each of these assessments, the CKBRC6 is fed by the round keys produced by the logistic map and the plain-images after extracting its header and divide the image stream into block starting from the upper left towards the right until the end of the stream. Due to the space limitation, a sample of each assessment is given to validate the obtained results; section 4.1 provides the visual inspection test, in 4.2 key space analysis is provided, information entropy analysis is given in 4.3, section 4.4 shows the results of cipher cycle analysis. Correlation coefficient and key sensitivity analyses are given in section 4.5 and 4.6 respectively.

4.1 Visual Testing

Visual testing constitutes one of the fundamental assessment for the quality of the cipher. The main aim of this test is to check if any visual information can be detected from the cipher-image. Here we apply the CKBRC6-32/32/20 to two different images: Lena and Cman each of 8-bits gray scale with 256×256 pixels. The outcomes of both encryption and decryption routines are given in Figure 2. It is obvious from these results, the cipher-images contain no visual information about the original image. Moreover, all cipher-images are indistinguishable even though the original images differ completely. Furthermore, results show the ability of CKBRC6 to decrypt images into their original state without information leakage.



a) Lena original image **b) Lena encrypted image** **c) Lena decrypted image**

d) Cman original image **e) Cman encrypted image** **f) Cman decrypted image**

Figure 2 Plain, encrypted and decrypted versions of Lena, Cman and Tiger images using CKBRC6

4.2 Key Space Analysis

The main aim of this analysis is to assess the invulnerability of CKBRC6 to statistical attack that tries all possible permutations of keys on a cipher-image sample until its plain version is obtained. Hence, the size of key space of a secure encryption scheme should be large to make such attack unattainable. CKBRC6 achieves this requirement by offering a large key space size ranging from (0-2040) bits. Thereby, an adversary requires 2^{4040} operations to determine the encryption key which make it infeasible. A justification for this argument can be drawn by calculating the computational load required to generate the key space using the-state-of-the art CPUs. Assuming that we use Intel Core i7 6950X whose MIPS (Mega instruction per Cycle) is 317,900 at 3.GHz. Then the total time required to generate key space of CKBRC6 with just 128 bits key length is:

$$\frac{2^{128}}{317{,}900 \times 10^6 \times 60 \times 60 \times 24 \times 365} = 4.177206 \times 10^{17}\ years \quad (2)$$

This is very long time that exceeds the useful lifetime of images.

4.3 Information Entropy Analysis

The concept of entropy in the context of information theory was defined to quantify the randomness of a message by measuring the expected amount of information conveyed in a message. In mathematical notation, let us consider an image with L pixel intensity scales whose values can be enumerated as $\{m_1, m_2, .., m_L\}$ and the probability at which m_i takes place in an image is denoted by $p(m_l); 1 \le l \le L$ then the entropy of the image is given in bit and computed as:

$$H(m) = \sum_{l=1}^{L} p(m_l) \log_2 \frac{1}{p(m_l)} \quad \text{bits} \quad (3)$$

This equation states that the maximum value of $H(m)$ is $\log_2 L$ can be attained only when pixel's intensities are all equiprobable, i.e., $p(m_l) = 1/L\,;\, \forall\, m_l \in \mathcal{M}$ which take place only in true random images.

Considering the fact that a good cipher should generate cipher-images that cannot be discerned from true random images in order to prevent entropy attacks and reduce the information leakage [32]. Therefore, entropy of encrypted-images should approach the maximum possible value. Here we evaluate the entropy of encrypted-image generated by CKBRC6 algorithm by applying it to the Lena image, compute the frequencies at which each ciphertext's pixel values are generated and then divided them into the total number of pixels to obtain $p(m_i)$. Substituting them into equation (3) yields $H(m) = 7.9977$ which is very close to the theoretical maximum value of random image with identical pixel value (i.e. 8). It is worth notating that this value is higher than its peer that can be achieved by original RC6 scheme which is 7.9899[33] which demonstrates the improvement of our proposal compared to RC6.

4.4 Cipher Cycle Analysis

Differential cryptanalysis aims to reveal some insightful relationships between the encrypted and plain images by figuring out the effects of a tiny change (e.g., just a single pixel) in plain-images on their corresponding cipher-images [32]. Hence one of the vital assessment for a cipher is to measure to what extent a minor change in plain-image can yield a completely different ciphered images. The common metric that used to quantify the immunity of a cipher to such attack is the Number of Pixel Change Rate (NPCR). NPCR evaluates the percentage of pixels that have been changed in cipher-images as a result of change a single pixel in plain-image. In order to describe this metric mathematically, let C and $\acute{C}$ be the image components of two encrypted images of the same plain-image except a single pixel difference. Moreover, let (i, j) be the pixel index of the i^{th}row and j^{th} column of these two images, then $NPCR$ can be calculated as:

$$NPCR = \frac{\sum_{i,j} D(i,j)}{N} \times 100\% \qquad (4)$$

Where N is the number of pixels and $D(i, j)$ is an indicator function that defined as:

$$D(i,j) = \begin{cases} 0\ if\ \ C(i,j) = \acute{C}(i,j) \\ 1\ if\ C(i,j) = \acute{C}(i,j) \end{cases} \qquad (5)$$

Applying $NPCR$ on two random images and computing the its average value yields:

$$\mathbb{E}[NPCR] = (1 - 2^{-L}) \times 100\% \qquad (6)$$

where L is the pixel intensity expressed in bits, take for instance two random images with identical gray depth of value 8 bits, i.e., $L = 8$ bits gives:

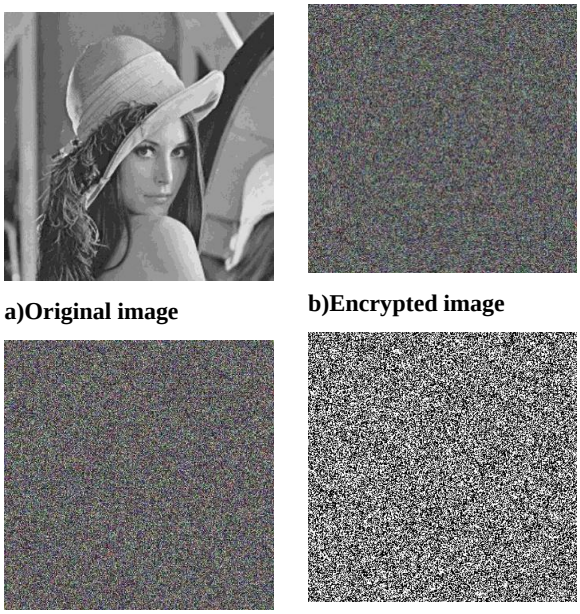
$$\mathbb{E}[NPCR] = 99.609375\% \qquad (7)$$

Assessment for security of CKBRC6 using NPCR metric has been conducted in this study by changing just a single bit in several widely-used images. Thereafter, we apply CKBRC6 to the two images and tabulated the values of $\mathbb{E}[UACI]$ in table 2.

Table2. Values of $\mathbb{E}[UACI]$ for some images

Image	$\mathbb{E}[NPCR]$ of CKBRC6	$\mathbb{E}[NPCR]$ of RC6
Lena	99.60273%	99.45888%
Cman	99.60752%	99.45798%

This table demonstrates that CKBRC6 can always produces higher readings compared to RC6 which is attributed mainly to the better diffusion process in CKEBRC6 that enables it to diffuse a minor change in original image to the entire cipher-images. Moreover, it can be seen of this results that the value $\mathbb{E}[NPCR]$ of CKBRC6 is very close to the highest values that can be obtained in true random image. Visualization for above results is shown in Figure. 3 using Lena image in 3-a and its encrypted image in 3-b. In 3-c the encrypted image of Lean with a single bit change is shows and finally in 3-d the difference between the two cipher-images (3-b) and (3-c) is shown. As it can be seen, this difference is big enough which demonstrates the high sensitivity of the CKBRC6 to a single bit change in original image.



a)Original image **b)Encrypted image**

c)Encrypted image with 1-bit difference in plain-image **d)difference image between (b) and (c)**

Figure 3 Plaintext sensitivity test with CKBRC6-32/32/20

4.5 Correlation Coefficient Analysis

Correlation coefficient is a statistical metrics that is used in the domain of image processing to quantify the degree at which a pair of adjacent pixels is related. Although it is usual that pixels of plain images are highly correlated due their visual contents a crypted image must sever these relations in order to prevent statistical attacks. The correlation coefficient (r_{xy}) of an image of N pixels can be computed as:

$$r_{xy} = \frac{\sum_{i=1}^{N} \mathbb{E}\big(x_i - \mathbb{E}(x)\big)\mathbb{E}\big(x_i - \mathbb{E}(y)\big)}{\sqrt{\Big(\sum_{i=1}^{N} x_i - \mathbb{E}(x)\Big)^2}\sqrt{\Big(\sum_{i=1}^{N} y_i - \mathbb{E}(y)\Big)^2}} \qquad (8)$$

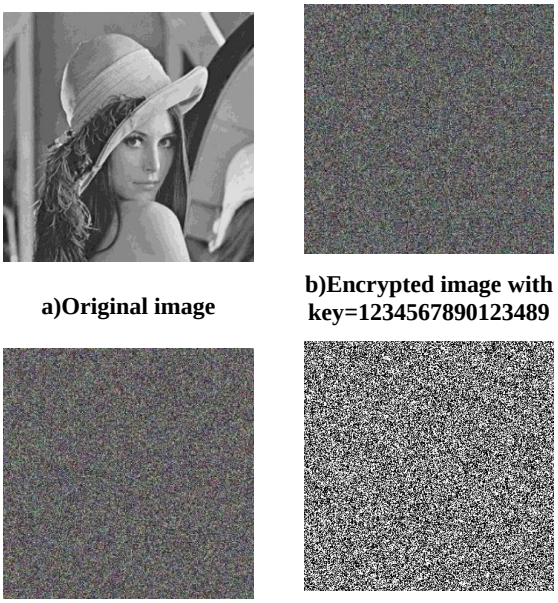
where x_i and y_i denote the values of two neighboring pixels in the image, and $\mathbb{E}(x)$ is their average values. In this work, 10^3 pairs of two vertically, horizontally and diagonally adjacent pixels are selected randomly and then equation (11) is used to measure r_{xy} before and after applying CKBRC6, table 3 shows the results of this test.

Table 3. Correlation coefficients in plain-image/cipher-image

Direction of adjacent pixels	r_{xy} of Plain-image	r_{xy} of Cipher-image
Horizontal	0.9921	0.0057
Vertical	0.9852	- 0.0015
Diagonal	0.9768	- 0.0074

4.6 Key Sensitivity Analysis

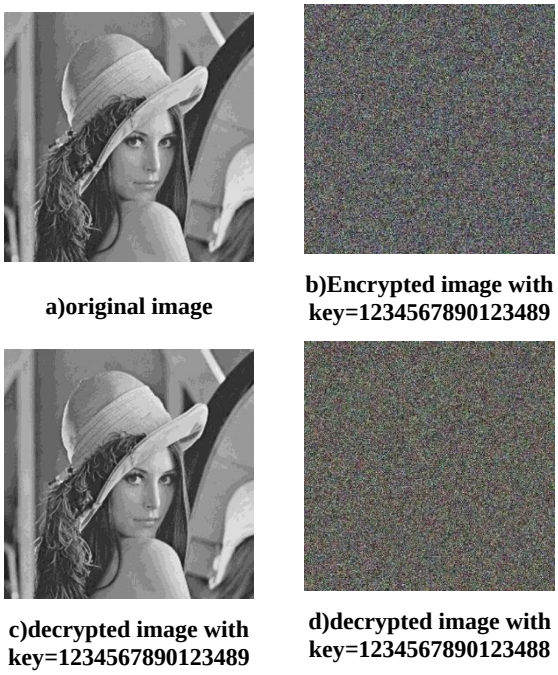
One of the most important metrics that is used to evaluate the secure cipher is to measure its sensitivity for a minor change in encrypted key. Typically, a tiny modification in encrypted key must produce a massive change in the outcomes [32]. Here we feed a 512×512 Lena image and a test ciphering key (i.e., 1234567890123489) to the CKBRC6. Thereafter, the same procedure is repeated but with a slight change in the key (i.e.,1234567890123488). The ciphered images are then compared to examine the difference between them. The results as depicted in Figure 4-b and c shows two encrypted images with the two keys, and the comparison between them as illustrated in Figure 4-d shows that 99% of the gray levels are dissimilar. Another assessment for the key sensitivity is presented in Figure 5, wherein Figure 5-a, the original image is shown in Figure 5-b, and the encrypted image with key (1234567890123489) is given and in 5-c the encrypted image is given. Finally, in Figure 5-d, the other key (1234567890123488) is used to decrypt the image. These results show that decryption processes fail to produce the original image when different key is used. This, in turn, highlights the high sensitivity of CKBRC6 to a minor change in the encrypted key.



a)Original image **b)Encrypted image with key=1234567890123489**

c)Encrypted image with key=1234567890123488 **d)Difference image between (b) and (c)**

Figure 4 key sensitivity analysis result 1 with CKBRC6-32/32/20



a)original image **b)Encrypted image with key=1234567890123489**

c)decrypted image with key=1234567890123489 **d)decrypted image with key=1234567890123488**

Figure 5 Key sensitivity test result 2 with CKBRC6-32/32/20

5 CONCLUSION

This paper proposed a chaotic key based RC6 block cipher (CKBRC6) as enhancement version for the original RC6 algorithm. The proposed scheme makes use of logistic map to overcome one of the major shortcomings of RC6. Experimental result verified the efficiency and superiority of the proposed CKBRC6 in terms of different security tests like visual inspection, correlation coefficient, and differential and sensitivity analysis.

6 REFERENCES

[1] "3.5 million photos shared every minute in 2016 | Deloitte UK", Deloitte United Kingdom, 2018. [Online]. Available: https://www2.deloitte.com/uk/en/pages/press-releases/articles/3-point-5-million-photos-shared-every-minute.html. [Accessed: 05- Jun- 2018].

[2] Zhang, X.; Wang, and Y. Zhang, "The visual internet of things system based on depth camera," *in Proceedings of the Chinese Intelligent Automation Conference (CIAC '13)*, vol. 255, pp. 447–455, Yangzhou, China, 2013.

[3] Chen, Chen, Ren, Yuzhuo, Kuo, C.-C. Jay, Big Visual Data Analysis, Scene Classification and Geometric Labeling, *Springer Briefs in Signal Processing*, 2016.

[4] "3.5 million photos shared every minute in 2016 | Deloitte UK", Deloitte United Kingdom, 2018. [Online]. Available: https://www2.deloitte.com/uk/en/pages/press-releases/articles/3-point-5-million-photos-shared-every-minute.html. [Accessed: 05- Jun- 2018].

[5] "Cyber Incident & Breach Trends Report - Online Trust Alliance", Online Trust Alliance, 2018. [Online]. Available: https://www.otalliance.org/system/files/files/initiative/documents/ota_cyber_incident_trends_report_jan2018.pdf. [Accessed: 05- Jun- 2018].

[6] Encrypted Traffic Analytics, Cisco Systems, 2018. [Online]. Available:https://www.cisco.com/c/dam/en/us/solutions/

collateral/enterprise-networks/enterprise-network-security/nb-09-encrytd-traf-anlytcs-wp-cte-en.pdf. [Accessed: 05- Jun- 2018].

[7] R. Rivest, M. Robshaw, R. Sidney, and Y.L. Yin, "The RC6 Block Cipher," NIST AES Proposal, 1998.

[8] R.L. Rivest, "The RC5 Encryption Algorithm," Fast Software Encryption, 2nd International Workshop Proceedings, *Springer-Verlag*, pp. 86–96, 1995.

[9] Miyaji A., Takano Y. "On the Success Probability of χ2-attack on RC6". In Boyd C., González Nieto J.M. (eds) Information Security and Privacy. ACISP 2005. Lecture Notes in Computer Science, vol 3574. *Springer, Berlin*, Heidelberg, 2007.

[10] H. Gilbert, H. Handschuh, A. Joux, and S. Vaudenay, "A Statistical Attack on RC6", FSE 2000, LNCS 1978(2000), *Springer-Verlag*, 64–74, 2000.

[11] L. Knudsen and W. Meier, "Correlations in RC6 with a reduced number of rounds", FSE 2000, LNCS 1978, *Springer-Verlag*, 94–108, 2000.

[12] A. Miyaji and M. Nonaka, "Cryptanalysis of the Reduced-Round RC6", ICICS 2002, LNCS 2513, *Springer-Verlag*, 480–494, 2002.

[13] N. Isogai, T. Matsunaka, and A. Miyaji, "Optimized χ2-attack against RC6", ANCS 2003, LNCS 2846, *Springer-Verlag*, 2003.

[14] S. Contini, R. Rivest, M. Robshaw, and Y. Yin, "The Security of the RC6 Block Cipher. v 1.0,", 1998.

[15] T. Shimoyama, M. Takenaka, and T. Koshiba, "Multiple linear cryptanalysis of a reduced round RC6," *FSE 2002*, LNCS 2365, Springer-Verlag, 76–88, 2002.

[16] Boyar, J.. "Inferring sequences produced by a linear congruential generator missing low-order bits." *Journal of Cryptology*, 1, 177–184, 1989.

[17] Brickell, E.F. and A.M. Odlyzko. "Cryptanalysis: A survey of recent results." Contemporary Cryptology: *The Science of Information Integrity*, 501–540, IEEE Press, Piscataway, NJ, 1992.

[18] Frieze, A.M., J. Hastad, R. Kannan, J.C. Lagarias, and A. Shamir. "Reconstructing truncated integer variables satisfying linear congruence." *SIAM Journal on Computing*, 17, 262–280, 1992.

[19] Krawczyk, H.. "How to predict congruential generators." *Journal of Algorithms*, 13, 527–545, 1992.

[20] Plumstead, J.B. "Inferring a sequence generated by a linear congruence." *Proceedings of the IEEE 23rd Annual Symposium on Foundations of Computer Science*. IEEE Press, New York, 153–159, 1982.

[21] Plumstead, J.B.. "Inferring a sequence produced by a linear congruence." Advances in Cryptology—CRYPTO'82, *Lecture Notes in Computer Science, eds*. D. Chaum, R.L. Rivest, and A.T. Sherman. Plenum Press, New York, 317–319, 1983.

[22] Stern, J. "Secret linear congruential generators are not cryptographically secure*." Proceedings of the IEEE 28th Annual Symposium on Foundations of Computer Science*. IEEE Press, New York, 421–426, 1987.

[23] G. Alvarez, S. Li, "Some basic cryptographic requirements for chaos-based cryptosystems", *Int. J. Bifurcation Chaos*, vol. 16, no. 8, pp. 2129-2151, 2006.

[24] Marcel Ausloos, Michel Dirickx, "The Logistic Map and the Route to Chaos: From the Beginnings to Modern Applications", *Springer Science & Business Media*. 2016.

[25] R. Bose, A. Banerjee, "Implementing symmetric cryptography using chaos functions", *Proc. 7th Int. Conf. Advanced Commun. Comp*. (ADCOM), pp. 318-321, 1999.

[26] Routo Terada and Eduardo T. Ueda. 2009. "A new version of the RC6 algorithm, stronger against χ^2 cryptanalysis", In *Proceedings of the Seventh Australasian Conference on Information Security* - Volume 98 (AISC '09), Ljiljana Brankovic and Willy Susilo (Eds.), Vol. 98. Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 47-52., 1998.

[27] G. H. Kim, J. N. Kim and G. Y. Cho, "An improved RC6 algorithm with the same structure of encryption and decryption," *11th International Conference on Advanced Communication Technology, Phoenix Park*, pp. 1211-1215, 2009.

[28] Mardiana, Fajrillah, Yuyun and Khair, Ummul. "modification of RC6 Block Cipher Algorithm on Digital Images", *Journal of Physics: Conference Series*. 930. 012047. 10.1088/1742-6596/930/1/012047, 2017.

[29] M. Hamdi, R. Rhouma, S. Belghith, "A very efficient pseudorandom number generator based on chaotic maps and S-box tables", *Int. J. Comput. Control Quantum Inform. Eng*, vol. 9, pp. 481-485, 2015.

[30] V. Patidar and K. K. Sud, Informatica 33, 441 (2009). S. Ahadpour, Y.R. Sadra, and Z. ArastehFard, *Int. J. Computer Science* Issues 9, 449 , 2012.

[31] Vaidyanathan, S., Volos, C., Pham, V., et al. "Analysis, adaptive control and synchronization of a novel 4-D hyperchaotic hyperjerk system and its SPICE implementation. *Archives of Control Sciences*, 25(1), pp. 135-158. Retrieved 14 Apr. 2018, from doi:10.1515/acsc-2015-0009.

[32] A. G. Radwan, S. H. AbdElHaleem, and S. K. Abd-El-Hafiz, "Symmetric encryption algorithms using chaotic and non-chaotic generators: A review," *J. Adv. Res*., vol. 7, no. 2, pp. 193–208, 2016.

[33] A. Shrivastava, L. Singh, "An Efficient RC6 based Image Cryptography to Enhance Correlation and Entropy", *International Journal of Computer Applications*, vol. 139, no. 1, pp. 42-49, 2016.