2005 International Symposium on Nonlinear
Theory and its Applications (NOLTA2005)
Bruges, Belgium, October 18-21, 2005

General Design Rules for Chaos-Based Encryption Systems

Kristina Kelber† and Wolfgang Schwarz‡

†Department of Electrical Engineering, University of Applied Sciences,
Friedrich-List-Platz 1, D-01069 Dresden, Germany
‡Department of Electrical Engineering, Dresden University of Technology
Mommsenstr. 13, D-01069 Dresden, Germany,
Email: kelber@et.htw-dresden.de, schwarz@iee1.et.tu-dresden.de

Abstract—During the past decade a large number of chaos-based encryption systems has been suggested and investigated. Several of these systems are not suitable for cryptographic applications as they are cryptographically weak. In this paper some general design rules for chaos-based encryption systems are derived based on strenghts and weaknesses of already suggested systems. To demonstrate the importance of each design rule an example system which does not obey this rule is given and its weakness is shown.

1. Motivation and objective of the work

During the past decade a large number of chaos-based encryption systems has been suggested and investigated (e.g. [1, 2, 3]). The idea behind is to use complex dynamics but simple mathematical descriptions and algorithms of chaotic systems for the purpuse of encryption. So the design of these systems has generally be done on symbol level and not - as in classical cryptography - on bit level. This approach has some general limitations concerning the cryptographic strength of the designed systems as pointed out in [4].

On the other hand for applications like encryption of image and audio data cryptographical requirements are often not so strong as for other applications. But the amount of data to process is very large and thus for classical cryptosystems the computational effort might be very high.

Therefore the scope of this paper is to derive some general design rules for chaos-based encryption systems according to strengths and weaknesses of already suggested systems. These design rules shall help to reduce or even to avoid cryptographical weaknesses of chaos-based encryption systems.

Signals of (analogue) chaotic circuits are often not exactly reproducible due to inevitable small changes in initial conditions or system parameters. As for most applications an exact recovery of the original data is required here considerations are focused on discrete-time encryption systems.

To show the importance of each design rule the following *example systems* are used:

System E1: *Image Encryption Scheme Based on 3D Chaotic Baker Map* [3, 5]

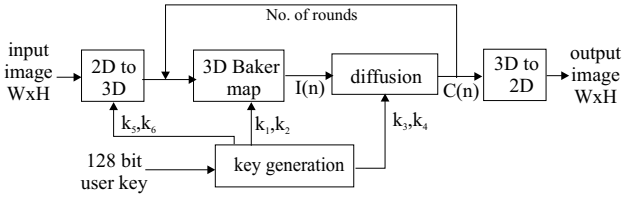

Figure 1: Image Encryption Scheme Based on 3D Baker Map

The system is designed to encrypt (a sequence of) RGB pixels (24 bit each) of an image. As shown in Fig. 1 it is based on an invertible 3D Baker map which permutes the position of the pixel values in a 3D cube and a diffusion operation

$$C(n) = \phi(n) \oplus \{[I(n) + \phi(n)] \ mod \ 2^{24}\} \oplus C(n-1) \quad (1)$$

where $I(n)$ represents the sequence of pixel values prior to diffusion and $C(n)$ afterwards. The sequence $\phi(n)$ is generated by the logistic map

$$\phi(n+1) = 4 \cdot \phi(n)(1 + \phi(n)) \quad (2)$$

For encryption a key of 128 bit is used which consists of six subkeys (k_1 to k_4: 24 bits, k_5, k_6: 16 bits). These bits directly correspond to six system parameters.

System E2: *Discrete-time chaos-based encryption system using modulo nonlinearity* [4, 6, 7, 8]

A general design of a self-synchronizing stream cipher is given in [6]. Here the simplest realisation of it (Fig. 2) is considered. It consists of a linear filter and just one piecewise-linear nonlinearity

$$mod(x) = x - 2 \cdot \left\lfloor \frac{x+1}{2} \right\rfloor \quad (3)$$

2. Methods of cryptoanalysis

The secrecy of a cryptosystems is determined by the minimum effort (e.g. resources, time), which is neccessary to break this system. So the aim of a design should

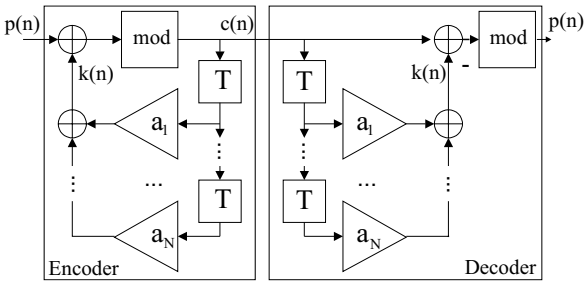

Figure 2: Discrete-time chaos-based encryption system using modulo nonlinearity

be a system which resists all known attacks (see i.e. [9]), such as brute-force attack (extended key search), chosen plain text attack, chosen cipher text attack, cipher text only attack and differential attack, as long as possible.

3. Design rules

3.1. System structure

For the design of chaos-based encryption systems there exist two general requirements:

- In order to retrieve the plain text during decryption the encryption has to be invertible.
- As for most implementations of (discrete-time) chaos-based encryption systems digital hardware is used chaotic maps have to be selected which preserve important properties (such as mixing or uniform probability density function of the generated signal) when digitized.

In classical cryptography there exist two approaches: block ciphers and stream ciphers. Block ciphers map (identical) blocks of plain text to (identical) blocks of cipher text. Therefore their basic mode of operation is called electronic codebook mode (ECB). From nonlinear dynamics point of view they can be considered as static nonlinear maps. Stream cipher process data sequences using dynamical systems.

In chaos-based cryptography both approaches are used as well. Chaotic block ciphers use a plain text block as initial condition and/or parameter of a chaotic map (e.g. in [1]). As chaotic maps are not invertible a suitable method of discretization of chaotic maps to invertible maps is required. A simple example is the 1D Baker map $B : I = [0, 1) \rightarrow I$

$$B(x) = 2 \cdot x - \lfloor 2 \cdot x \rfloor. \quad (4)$$

Discretization of it to $I = [0, 2N)$ leads to the invertible map

$$B_d(x) = \left\{ \begin{array}{ll} 2x & 0 \leq x < N \\ 2x - (2N-1) & N \leq x < 2N \end{array} \right\}. \quad (5)$$

A general method of discretization is shown in [2] and examples of suitable maps as well as their application to image encryption are presented e.g. in [2, 3].

For stream ciphers an invertible combining function $\varphi(.,.)$ is used which combines plain text symbols $p(n)$ and key symbols $k(n)$ to cipher text symbols $c(n)$ (Fig. 3).

$$p(n) \rightarrow \boxed{\varphi(.,.)} \rightarrow c(n) \rightarrow \boxed{\varphi^{-1}(.,.)} \rightarrow p(n)$$
$$\uparrow \quad\quad\quad\quad\quad\quad \uparrow$$
$$k(n) \quad\quad\quad\quad\quad\quad k(n)$$

Figure 3: Combining function $\varphi(.,.)$ and its inverse

In chaos-based systems discretized combining operations have to be invertible as well. Examples of balanced combining functions are binary XOR or modulo addition

$$c(n) = mod(p(n) + k(n))$$

as in system E2. An advantage of this so-called inverse system approach [10] is that a chaos-based key stream generator can be used which does not need to be invertible and which can be designed according to prescribed characteristics of the key stream such as uniform distribution [6].

- Rule 1: *Do either use suitable chaotic maps which preserves important properties during discretization (for block cipher) or a balanced combining function and a suitable key stream generator (for stream cipher).*

3.2. Key space

In order to prevent a successful extended key search the key space has to be very large and a key space reduction has to be made impossible. To achieve this rules 2 to 6 are important.

- Rule 2: *Do use large key space!*

This can be achieved using a large block length as well as a large number and a suitable precision of corresponding sensitive system parameters. For example, when encrypting images one could use a RGB pixel (24 bits) as one symbol instead of each single byte. This leads to a larger number of bits for each parameter as e.g. for system E1 in [3].

- Rule 3: *Do not use initial condition of an inverse system as part of the key!*

In system E1 the key k_4 (24 bit integer) is used as initial condition $C(0)$ for the diffusion operation (1). In decryption process the inverse operation [3] is described by

$$I(n) = [\phi(n) \oplus C(n) \oplus C(n-1) - \phi(n)] \; mod \; 2^{24}. \quad (6)$$

The key k_4 only influences the value of the first pixel if one round of encryption is used. As an example Fig. 4d) shows a decrypted image after changing the subkey k_4 from 012 to $01A$.

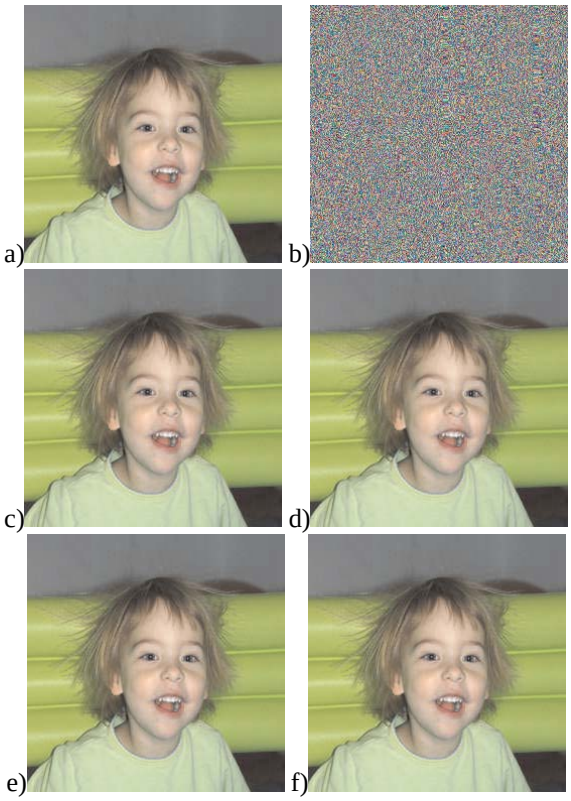

Figure 4: Test Bitmap ($W \cdot H = 256 \cdot 256$) for system E1 using key 1234567890123456 and one round of encryption: a) original, b) encoded and decoded using c) correct key, d) $k_4 = 012 \rightarrow 01A$, e) $k_5 = 34 \rightarrow 3A$ and f) $k_6 = 56 \rightarrow 5A$

- Rule 4: *Do avoid simple permutation of identical system parameters!*

In system E1 [3] a key of 128 bit is used. These bits directly correspond to six system parameters. The subkeys k_5 and k_6 (16 bit integers) are used for permutation of the prime factors of width W and height H of the original image in order to determine the size of the 3D cube. If $W \cdot H$ is a power of 2, then this permutation has no influence at all and thus the key space is reduced by 32 bits. An example is shown in Fig. 4 e) and f).

- Rule 5: *Do use same precision for subkey values and corresponding system parameters!*

In system E1 the subkey k_3 is used as initial condition $\phi(0) \in [0, 1)$ for the generation of the sequence $\phi(n)$ (2) in the diffusion operation (1). Conversion of the 24 bits of k_3 to a floating point number may reduce the key space for floating point implementations when short mantissas are used.

- Rule 6: *Do use complex input key transformation!*

Often it is possible to analyze the behavior of a subsystem in dependence of a certain parameter. When key bits are directly used as parameter bits the determination of this parameter reduces the key space by the corresponding number of bits. E.g. in system E1 [3] the analysis of subkeys as discussed in connection with rules 3 and 4 reduces the key space from 2^{128} to 2^{72} bits. Using a complex transformation of the key bits to the parameter bits avoids this problem. E.g. in [11] it is suggested to use the key bits as parameters and initial values of another chaotic system, iterate it a certain number of times and then derive the system parameters of the encryption system from the states of the other chaotic system.

A transformation is particularly suitable which changes approximately half of the parameter bits when one key bit is changed as it is often done in classical cryptography.

3.3. Nonlinearity and dynamics

In many cases for chaos-based encryption systems simple chaotic maps, such as logistic map, piecewise linear maps or even Markov maps, are used. One reason is that the properties of these maps, especially statistial ones, can be calculated or even designed. An example of such a design is given e.g. in [6]. From a cryptographical point of view these simple nonlinearities are in general easy to analyse. In order to increase the amount of work nesseccary for all cryptographical attacks, the nonlinearity has to be as complex as possible (taking into account computational effort and time requirements) and its analysis has to be as difficult as possible. To achieve this rules 7 to 10 are important.

- Rule 7: *Do use a dynamical system!*

Encryption by a static nonlinear transformation F_k (electronic codebook mode, ECB) leads for identical plain text blocks to identical cipher text blocks. Then the codebook is easy to analyse, especially if the block size is very small (often just a single symbol in chaos-based encryption systems). Furthermore an image encrypted in ECB mode still provides lots of information about the plain text image as the redundancy is very high in images. An examples using the Data Encryption Standard (DES) in ECB mode is given in [8].

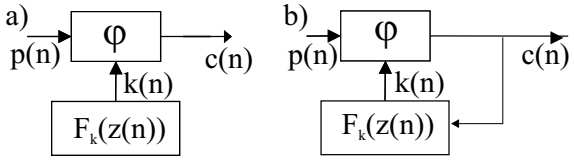

Figure 5: Modes of operation: a) CFB, b) OFB

This disadvantages of static encoding (ECB mode) can be avoided by using a dynamical system $F_k(z(n))$ where the cipher text block $c(n)$ depends on the internal state $z(n)$ of this dynamical system too. In chaos-based cryptography there exist two approaches: synchronized scheme (corresponding to cipher feedback mode, CFB, Fig. 5a),

in classical cryptography) and self-synchronizing scheme (corresponding to output feedback mode, OFB, Fig. 5b)).

- Rule 8: *Do use complex nonlinearities!*

Simple nonlinearities can easily be analyzed from an cryptographical point of view. This is demonstrated for the system E2 (as shown in Fig. 2) in [7] using different methods of cryptoanalysis. Combining several of these basic nonlinearities increases the complexity of the nonlinearity. Especially for self-synchronizing stream ciphers several suitable nonlinearities can be added to the dynamical subsystem $F_k(z(n))$ (see Fig. 5b) as it is suggested e.g. for the system E2 in [8]. Then is becomes much more complicated to analyze the nonlinearity even by a chosen plain text or a differential attack and statistical properties of the encrypted signal can still be calculated or designed.

- Rule 9: *Do modify nonlinearities depending on key and signal values!*

Analysis of a static nonlinearity is much easier then analysis of a nonlinearity which is changing with time. Parameterizing the nonlinearity in dependence of the key is necessary but modifying it in time, e.g. depending on current signal values is much better. The latter can be done by modifying either basic nonlinearities or the combination of them. In a digital realization of the system E2 (Fig. 2) e.g. the carry-in bit of the adders can be modified in dependence on some signal value or bit as suggested in [8].

- Rule 10: *Do use several rounds of operation for block ciphers!*

Imagine the system E1 is operated with just one round. Then the key $k_4 = C(0)$ can be determined by a chosen plain text attack using a black image (represented by a matrix of zeros) as input. As the 3D Baker map only performs a permutation of the position of the pixel values the input stream $I(n)$ for the diffusion operation (1) will be zero as well. In this case the diffusion operation will be simplified to

$$C(n) = \phi(n) \oplus \phi(n) \oplus C(n-1) = C(n-1) \qquad (7)$$

and thus it holds $C(n) = C(0) = k_4$ for all n.

Furthermore the parameters of the 3D Baker map (which are equivalent to k_1 and k_2) can be determined by a differential attack. For it a gray input image (Fig. 6a) is used and in each analysis step just one pixel is modified (e.g. Fig. 6b). Because the 3D Baker map [3] is piecewise linear its parameters can be reconstructed based on the relationship between the positions of the modified pixel in the input and the first modified pixel in the encoded image (see e.g. Fig. 6c, using offset 128 for presentation). In a second round of operation the 3D Baker map will permute the sequence $C(n)$ and thus this analysis will not be possible any more.

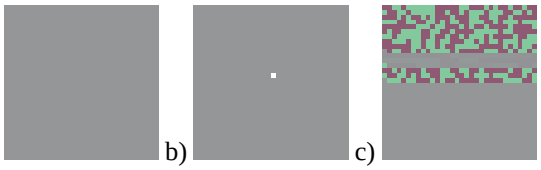

Figure 6: Example images: a) gray, b) one pixel changed to white, c) difference of encoded images of a) and b) (Offset 128)

4. Summary

In this paper some general design rules for chaos-based encryption systems have been presented. These rules concern system structure, key space as well as nonlinearity and dynamics of the encryption system. They should be helpful to reduce or even to overcome cryptographical weaknesses of such systems. Importance of these design rules has been demonstrated on two chaos-based example systems.

Acknowledgement

The authors are grateful to Frank Dachselt for stimulating discussions.

References

[1] T. Habutsu, Y. Nishio, I. Sasase, S. Mori: A Secret Key Cryptosystem by Iterating a Chaotic Map, *Proc. Eurocrypt*, 127-139, 1991.

[2] J. Fridrich: Symmetric Ciphers Based on Two-dimensional Chaotic Maps, *Int. J. Bif. & Chaos*, vol. 8, 1259-1284, 1998.

[3] Y. Mao, G. Chen, S. Lian: A Novel Fast Image Encryption Scheme Based on 3D Chaotic Baker Maps, *Int. J. Bifurcation & Chaos*, June, 2003.

[4] F. Dachselt, W. Schwarz. Chaos and Cryptography, *IEEE Trans. Circ. & Syst.-I*, vol. 48(12):1498-1509, 2001.

[5] Y. Mao, G. Chen: Chaos-Based Image Encryption, in E. Bayro-Corrochano (ed.), *Handbook of Computational Geometry for Pattern Recognition, Computer Vision, Neural Computing and Robotics*, Springer, 2003.

[6] M. Götz, K. Kelber, W. Schwarz. Discrete-time chaotic encryption systems - Part I: Statistical design approach, *IEEE Trans. Circ. & Syst.-I*, vol. 44(10), 963–969, 1997.

[7] F. Dachselt, K. Kelber, W. Schwarz. Discrete-time chaotic encryption systems - Part III: Cryptographical analysis, *IEEE Trans. Circ. & Syst.-I*, vol. 45(9), 983–988, 1998.

[8] K. Kelber: Application of nonlinear dynamical systems to image encryption, *Proc. NDES 2004*, Evora, Portugal, 186-189, 2004.

[9] B. Schneier: *Applied Cryptography*, Wiley, 1996.

[10] U. Feldmann, M. Hasler, and W. Schwarz. Communication by chaotic Signals: The inverse Systems Approach, *Int. J. Circuit Theory Appl.*, 24(5):551-579, 1996.

[11] G. Chen, Y. Mao, C.K. Chui: A symmetric image encryption scheme based on 3D chaotic cat map, *Chaos, Solitons and Fractals*, vol. 21, 749-761, 2004.