NOTICE: This is the author's version of a work that was accepted by *Communications in Nonlinear Science and Numerical Simulations* in February 2009. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version has been published in *Physics Letters A*, vol. 373, no. 37, pp. 3398-3400, 2009, Elsevier. DOI: 10.1016/j.physleta.2009.07.035.

Comments on "Modified Baptista type chaotic cryptosystem via matrix secret key" [Phys. Lett. A 372 (2008) 5427]

Rhouma Rhouma*[,a], Ercan Solak[b], David Arroyo[c], Shujun Li[d], Gonzalo Alvarez[c], Safya Belghith[a]

[a] *Syscom Laboratory, Ecole Nationale d'Ingénieurs de Tunis, 37, Le Belvédère 1002 Tunis, Tunisia*
[b] *Department of Computer Science and Engineering, Isik University, Istanbul, Turkey*
[c] *Instituto de Física Aplicada, Consejo Superior de Investigaciones Científicas, Serrano 144—28006 Madrid, Spain*
[d] *Fachbereich Informatik und Informationswissenschaft, Universität Konstanz, Fach M697, Universitätsstraße 10, 78457 Konstanz, Germany*

Abstract

In this comment, we analyze a recently proposed Baptista-like cryptosystem and show that it is not invertible. Others weaknesses are also reported. A modified version of this cryptosystem is proposed to show how to overcome the non-invertibility.

Key words: Baptista-like cryptosystem, chaos-based cryptography, cryptanalysis, logistic map, skew tent map.

1. Introduction

In [1] M.S. Baptista proposed a chaos-based cryptosystem. The Baptista system uses the logistic map to generate chaos. A subset of the phase space of the logistic map, $[X_{\min}, X_{\max}] \subset [0, 1]$, is divided into 256 equal subintervals and each 8-bit plaintext character is assigned to one subinterval. The encryption process of a given plaintext character consists of iterating the logistic map until the state reaches the subinterval assigned to that character. The ciphertext corresponding to the plaintext character is the number of iterations.

Since Baptista's original proposal, many variants were proposed to enhance the performance of the original cryptosystem [2, 3, 4, 5]. However, most of these modified Baptista-like cryptosystems have been cryptanalyzed [6, 7, 8].

Recently, a new variant of the original Baptista cryptosystem was proposed in [9]. In this paper, we analyze this new variant and demonstrate that it is not invertible. We also show that the new variant suffers from the same weaknesses as the original Baptista cryptosystem.

The organization of the paper is as follows. Section 2 gives a brief description of the cryptosystem under study. In Section 3, we demonstrate the non-invertibility of the cryptosystem through a simple example. In Section 4, it is shown that the cryptosystem has some drawbacks when the underlying chaotic map is not selected properly. Section 5 shows how the key space can be drastically reduced by a partial key recovery attack. Section 6 demonstrates that there is a link between the size of the distortion matrix and

*Corresponding author: Rhouma Rhouma (rhoouma@yahoo.fr).

the encryption time, which leads to a possible timing attack. In Section 7, we discuss how to modify the cryptosystem under study to make it invertible. In the last section, the main findings reported in this paper are summarized.

2. Brief description of the cryptosystem

The cryptosystem proposed in [9] transforms a plaintext into a ciphertext using the ergodic property of a certain chaotic map $f_\lambda : I \to I$, for $\lambda \in J$, $I, J \subset \mathbb{R}$. The phase space I is divided into N disjoint intervals, where N is the cardinality of the alphabet of plaintexts. Each of those intervals is associated with one symbol in the alphabet of plaintexts. For a plaintext $P = p_1 p_2 \dots p_m$ of length m, with $p_i \in \{1, 2, \dots, N\}$, the encryption procedure comprises the following steps:

1. Group the plaintext into a set of vectors B_j of length $k \in \mathbb{N}$, where $j = 1, 2, \dots, m/k$.
2. Multiply each vector B_j by a $k \times k$ distortion matrix A of integers to get a new vector Y_j as follows: $Y_j = A \cdot B_j$, $j = 1, \dots, m/k$.
3. Concatenate elements of all the vectors Y_j to obtain a sequence of integers $z_1, z_2, \dots, z_m$.
4. For each z_i, iterate the chaotic map z_i times and obtain the symbol u_i associated with the interval in which the final value lies in. Each time, start the iterations from the last final value.
5. Encrypt the symbols $u_1, \dots, u_m$ via the original Baptista method [1] starting from the original initial condition x_0.

The secret key of the cryptosystem under study is composed of two subkeys, i.e., $k = (k_1, k_2)$. The first subkey k_1 is the initial condition x_0 and k_2 is the distortion matrix A. The authors of [9] propose two possible chaotic maps to be used in the encryption process. The first proposed map is the logistic map, which is defined by

$$x_{n+1} = f_\lambda(x_n) = \lambda x_n (1 - x_n), \tag{1}$$

where the phase space is $I = [0, 1]$ and $\lambda = 4$. The other chaotic map is the skew tent map:

$$x_{n+1} = \begin{cases} x_n/\lambda, & \text{for } 0 < x \leq \lambda, \\ (1 - x_n)/(1 - \lambda), & \text{for } \lambda < x < 1, \end{cases} \tag{2}$$

where $I = [0, 1]$ and $\lambda \in (0, 1)$ is a public parameter.

The decryption procedure is roughly the inverse of the encryption procedure. Please refer to [9] for more details.

3. Non-invertible encryption procedure

Any encryption scheme must be designed in such a way that the recovering of the plaintext from ciphertext can be performed when the secret key is known. However, this is not the case for the cryptosystem described in [9]. Indeed, Step 4 of the encryption procedure does not implement a one-to-one operation. The operation transforms m integers $\{z_i\}$ (numbers of chaotic iterations) into m symbols $\{u_i\}$ in a smaller set (the plaintext alphabet). Apparently, two different numbers of chaotic iterations may lead the chaotic orbit to the same interval and thus the same symbol u_i. In other words, in the decryption procedure, if one meets a symbol u_i, one has no clue on how to determine the value of z_i. If one simply picks the smallest number of iterations which corresponds to u_i, which may not be the correct value, then the recovered plaintext will be wrong.

In the following, we illustrate this non-invertibility problem with a simple example. Assume that the plaintext has 4 different symbols $\{s_1 = 0, s_2 = 1, s_3 = 2, s_4 = 3\}$. In this case, the interval $I = [0, 1]$ is divided into 4 equal subintervals corresponding to the four symbols. That is, 0 corresponds to $[0, 0.25)$, 1 corresponds $[0.25, 0.5)$, and so on. Starting from $x_0 = 0.232323$ and iterating the logistic map with $\lambda = 4$, we obtained the following sequence of symbols according to the subintervals visited by each chaotic state x_i:

$$2, 3, 2, 3, 0, 1, 3, 0, 0, 0, 0, 2, 3, 0, 1, 3, 1, 3, \ldots$$

Let the plaintext be $P = \{s_1, s_2\} = \{0, 1\}$ and the distortion matrix $A = \begin{bmatrix} 2 & 3 \\ 1 & 2 \end{bmatrix}$. Then, the result of the distortion process is $\begin{bmatrix} 2 & 3 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$. By iterating the logistic map three times and then two times, we can get $\{s_3, s_1\} = \{2, 0\}$, and the final ciphertext will be $C = \{1, 4\}$. To decrypt the ciphertext, iterate the logistic map from x_0 one time and then four times, we will find the symbols $\{s_3, s_1\} = \{2, 0\}$. Iterate the logistic map until its chaotic orbit falls into the subinterval associated with the symbol $s_3 = 2$. Then, continue the iteration until the chaotic orbit falls into the subinterval associated with the symbol $s_1 = 0$. Finally we get $\{1, 4\}$. Do the inverse distortion operation, we have $A^{-1} \begin{bmatrix} 1 \\ 4 \end{bmatrix} = \begin{bmatrix} 2 & -3 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 4 \end{bmatrix} = \begin{bmatrix} -10 \\ 7 \end{bmatrix}$. Since -10 and 7 do not correspond to any symbols in the plaintext alphabet, the cryptosystem fails. Even if we map -10 and 7 to $\{0, 1, 2, 3\}$ via mod4, we will get $\{2, 3\}$, which is still wrong. Now we see the cryptosystem under study is not invertible and the original plaintext cannot be always recovered.

4. Problems arising from the selection of the chaotic map

Baptista's cryptosystem encrypts each plaintext symbol through a searching process along the orbit of a chaotic map. Originally the chaotic map used for encryption was the logistic map. However, the efficiency of the cryptosystem requires that the time necessary to locate the plaintext in a given orbit is independent of the value of the plaintext. This implies that the chaotic map involved should have a uniform invariant density function. Since the logistic map does not have a uniform invariant density function, it should be avoided in the searching-based chaotic cipher[1]. Good alternatives to the logistic map are piecewise linear chaotic maps [10] (such as the skew tent map), since they possess a uniform invariant density function for all possible values of the control parameter.

5. Partial key recovery attack

In the context of a secure and robust encryption system it is assumed that the partial knowledge of the key does not reveal information about the rest of the key and, as a result, the cryptosystem performance is not harmed [11, Rule 7]. As mentioned above, the secret key of the cryptosystem described in [9] consists of two subkeys, the initial condition x_0 ($k_1 = x_0$) used in the iteration of the chaotic map and the distortion matrix A ($k_2 = A$). If one knows the value of x_0, then a known-plaintext attack can be used to infer the matrix A. Indeed, if x_0 is known then Steps 4 and 5 of the encryption procedure are canceled. Since Step 3 does not depend on any subkey, it can also be skipped. Then one has both Y_j and B_j. With k different values of B_j and the corresponding values of Y_j, one can get two $k \times k$ matrices B and Y which satisfy $AB = Y$. If the rank of A is k, one can immediately derive $A = YB^{-1}$. If the rank of B is less than k, more (B_j, Y_j) pairs are needed to find k values of B_j which form a full-rank matrix. On the other hand, if the matrix A is known, then the cryptosystem can be attacked using any of the strategies explained in [8]. Therefore, the key space is drastically reduced to be the sum of the key spaces corresponding to the two subkeys, rather than their product.

6. Efficiency of the distortion process

The authors of [9] introduce the distortion matrix A in order to avoid the keystream recovery attack described in [7, 8]. Certainly, if the link between the symbols and the intervals is not known, then the

[1] In the original Baptista's cryptosystem, only the middle part of the whole phase space is used for mapping plaintext symbols to sub-intervals of the phase space. Since the middle part of the logistic map's invariant density function is relatively smooth, this can mitigate but not essentially solve the problem about non-uniformity.

keystream can not be recovered. Nevertheless, if the output of the distortion process is the number of times to iterate the chaotic map, it means that the encryption/decryption time depends on the value of the elements of A. In other words, the encryption/decryption time changes with the subkey k_2, which could make it possible to estimate the value of the key through a timing attack. Therefore, it is highly recommendable to establish another distortion procedure as the one described in [12]. In this case, the plaintext is encrypted from its binary codification, and each interval of the phase space is assigned in a random way to either a 0 or 1 bit.

7. Making the cryptosystem invertible

In this section, we show how the cryptosystem under study can be modified to be invertible. Note that we do not intend to improve the security of the original cryptosystem, but simply try to show how the non-invertibility problem can be overcome.

Since the non-invertibility problem is due to the chaotic iterations in Step 4 of the original cryptosystem, if we remove the chaotic iterations from Step 4 and adjust other steps accordingly, we will be able to get an invertible cryptosystem. The modified encryption procedure will comprise the following steps:

1. Group the plaintext into a set of vectors B_j of length $k \in \mathbb{N}$, where $j = 1, 2, \ldots, m/k$.
2. Do the following distortion operation for each B_j: $Y_j = (A \cdot B_j) \bmod N$, where A is a $k \times k$ distortion matrix and N is the cardinality of the plaintext alphabet.
3. Concatenate elements of all the vectors $Y_1, Y_2, \ldots, Y_{m/k}$ to obtain a sequence of integers $z_1, z_2, \ldots, z_m$.
4. Convert $z_1, z_2, \ldots, z_m$ to a symbol sequence $u_1, u_2, \ldots, u_m$ according to the map between the integers $\{0, \ldots, N-1\}$ and the N plaintext symbols.
5. Encrypt the symbols $u_1, u_2, \ldots, u_m$ via the original Baptista method [1].

The decryption procedure is straightforward, so we omit it here.

It is clear that the modified cryptosystem is just a simple combination of secret matrix computation (Step 2) and the original Baptista's chaos-based cryptosystem (Step 5). Since both steps are invertible, the whole encryption process is also invertible. Taking the same example in Section 3, we can show more clearly how the non-invertibility problem is solved. Let the plaintext be $P = \{s_1, s_2\} = \{0, 1\}$. The result of the distortion process is $\left(\begin{bmatrix}2 & 3\\ 1 & 2\end{bmatrix}\begin{bmatrix}0\\1\end{bmatrix}\right) \bmod 4 = \begin{bmatrix}3\\2\end{bmatrix}$, which corresponds to a symbol sequence $\{s_4, s_3\}$. Finally, iterate the chaotic map and we will get the final ciphertext $C = \{2, 1\}$. Decrypting the ciphertext with the original Baptista's method will lead to the same symbol sequence $\{s_4, s_3\} = \{3, 2\}$. Performing the inverse distortion process, we have $\left(A^{-1}\begin{bmatrix}3\\2\end{bmatrix}\right) \bmod 4 = \left(\begin{bmatrix}2 & 1\\ 3 & 2\end{bmatrix}\begin{bmatrix}3\\2\end{bmatrix}\right) \bmod 4 = \begin{bmatrix}0\\1\end{bmatrix}$. Now we get the original plaintext $\{s_1, s_2\} = \{0, 1\}$ back.

8. Conclusions

In this paper, we analyzed a recent chaotic encryption proposal. We demonstrated that the proposal involves a non-invertible encryption transformation. We also showed that the algorithm suffers from some other structural weaknesses which lead to more security problems.

Acknowledgments

The work described in this paper was partially supported by *Ministerio de Educación y Ciencia of Spain*, research grant SEG2004-02418, *CDTI, Ministerio de Industria, Turismo y Comercio of Spain* in collaboration with Telefónica I+D, Project SEGUR@ with reference CENIT-2007 2004, *CDTI, Ministerio de Industria, Turismo y Comercio of Spain* in collaboration with SAC, project HESPERIA (CENIT 2006-2009), and *Ministerio de Ciencia e Innovación of Spain*, project CUCO (MTM2008-02194). Shujun Li was supported by a fellowship from the Zukunftskolleg of the Universität Konstanz, Germany, which is part of the "Exzellenzinitiative" Program of the DFG (German Research Foundation). Ercan Solak was supported by the The Scientific and Technological Research Council of Turkey (TÜBİTAK) under Project No. 106E143.

References

[1] M. Baptista, Cryptography with chaos, Physics Letters A 240 (1998) 50.

[2] W.-K. Wong, L.-P. Lee, K.-W. Wong, A modified chaotic cryptographic method, Computer Physics Communications 138 (3) (2001) 234–236.

[3] K. W. Wong, A fast chaotic cryptographic scheme with dynamic look-up table, Physics Letters A 298 (4) (2002) 238–242.

[4] K.-W. Wong, A combined chaotic cryptographic and hashing scheme, Physics Letters A 307 (5-6) (2003) 292–298.

[5] K.-W. Wong, S.-W. Ho, C.-K. Yung, A chaotic cryptography scheme for generating short ciphertext, Physics Letters A 310 (1) (2003) 67–73.

[6] G. Alvarez, F. Montoya, M. Romera, G. Pastor, Cryptanalysis of dynamic look-up table based chaotic cryptosystems, Physics Letters A 326 (2004) 211–218.

[7] G. Alvarez, F. Montoya, M. Romera, G. Pastor, Keystream cryptanalysis of a chaotic cryptographic method, Computer Physics Communications 156 (2004) 205–207.

[8] G. Alvarez, F. Montoya, M. Romera, G. Pastor, Cryptanalysis of an ergodic chaotic cipher, Physics Letters A 311 (2003) 172–179.

[9] M. Ariffin, M. Noorani, Modified Baptista type chaotic cryptosystem via matrix secret key, Physics Letters A 372 (2008) 5427–430.

[10] S. Li, G. Chen, X. Mou, On the dynamical degradation of digital piecewise linear chaotic maps, International Journal of Bifurcation and Chaos 15 (10) (2005) 3119–3151.

[11] G. Alvarez, S. Li, Some basic cryptographic requirements for chaos-based cryptosystems, International Journal of Bifurcation and Chaos 16 (8) (2006) 2129–2151.

[12] F. Huang, Z.-H. Guan, Cryptosystem using chaotic keys, Chaos, Solitons and Fractals 23 (2005) 851–855.