Construction of a convolutional code based symmetric cryptosystem[1]

JOAN-JOSEP CLIMENT[(a)(b)], FRANCISCO FERRÁNDEZ[(a)(b)], VIRTUDES TOMÁS[(b)2]
[(a)]Institut d'Investigació Informàtica.
[(b)]Departament de Ciència de la Computació i Intel·ligència Artificial
Universitat d'Alacant
Campus de Sant Vicent del Raspeig
Ap. 99 E-03080 Alacant
SPAIN

Abstract: In this article we construct a concatenation of globally invertible convolutional codes based symmetric cryptosystem. At each step in the concatenation we propose two different systems choosing among them depending on the previous input weight. We describe the encryption and decryption processes and analyze the security of the proposed cryptosystem.

Key-words: Symmetric cryptosystem, convolutional code, state-space representation, concatenation.

1 Introduction

Symmetric cryptosystems [7, 10] are used to solve communication problems over insecure channels. A lot of encryption standards have being designed using this scheme (DES, AES, etc.) but sometimes they are not very practical. In addition, security of most of them has being questioned because of the new technologies improvements and the research of new cryptographic protocols. For this reason, it is a growing need of constructing new efficient symmetric cryptosystems.

Though coding theory and cryptography may have different purposes, it is posible to join them to develop cryptosystems, such as the block code based public key cryptosystem developed by McEliece [5].

In this article we propose a new private key cryptosystem based on convolutional codes and their concatenation.

Convolutional codes [1, 3, 6] are an specific class of error correcting codes that generalize block codes in a natural way. These codes can be represented as time-invariant discrete linear systems over a finite field [8].

For this work we have chosen the class of globally invertible codes [2, 4], since they allow us to uniquely decrypt the received sequences. In addition, we introduce some changes in the states of the systems in order to avoid time invariance and get a more dynamical model. About this new scheme it is possible to emphasize its simply and fast encryption and decryption processes.

The article is organized as follows. In section 2 we introduce some basic concepts about convolutional codes and the notation we will use. In section 3 we describe the presented cryptosystem emphasizing the steps to encrypt and decrypt a sequence. In section 4 we present a possible exhaustive key research and a known-plaintext attack. Finally, we provide some conclusions in Section 5.

2 Preliminaries

In this section we denote by $\mathbb{F}$ a finite field. A convolutional code $\mathcal{C}$ with rate k/n is a submodule of $\mathbb{F}^n[z]$ (see [9, 11]) that can be described as

$$\mathcal{C} = \{ \boldsymbol{v}(z) \in \mathbb{F}^n[z] \ : \\ \boldsymbol{v}(z) = G(z)\boldsymbol{u}(z) \text{ with } \boldsymbol{u}(z) \in \mathbb{F}^k[z] \}$$

where $\boldsymbol{u}(z)$ is the information vector or information word, $\boldsymbol{v}(z)$ is the code vector or code word and $G(z)$ is an $n \times k$ polynomial matrix with rank k called generator or encoding matrix of $\mathcal{C}$.

We call complexity of $\mathcal{C}$ the integer

$$\delta = \sum_{i=1}^{k} \nu_i$$

with ν_i being the maximum degree of the i-th column of $G(z)$.

McElice [6] describe a rate k/n convolutional code $\mathcal{C}$ of complexity δ by means of the system

$$\left.\begin{aligned} \boldsymbol{x}_{t+1} &= A\boldsymbol{x}_t + B\boldsymbol{u}_t \\ \boldsymbol{v}_t &= C\boldsymbol{x}_t + D\boldsymbol{u}_t \end{aligned}\right\} \quad t = 0, 1, 2, \ldots, \gamma \qquad (1)$$

[1] This work was partially supported by Spanish grant MTM2005-05759.

[2] The work of this author was partially developed during a stay at the University of Zurich and was supported by the Departament de Ciència de la Computació i Intel·ligència Artificial of the Universitat d'Alacant. The work of this author was also supported by a grant of the Vicerectorat d'Investigació, Desenvolupament i Innovació of the Universitat d'Alacant for PhD students.

$$\boldsymbol{x}_0 = 0, \quad \boldsymbol{x}_{\gamma+1} = 0$$

where $A \in \mathbb{F}^{\delta\times\delta}$, $B \in \mathbb{F}^{\delta\times k}$, $C \in \mathbb{F}^{k\times\delta}$ and $D \in \mathbb{F}^{k\times k}$ with

$$\begin{aligned}\boldsymbol{u}(z) &= \boldsymbol{u}_0 + \boldsymbol{u}_1 z + \boldsymbol{u}_2 z^2 + \cdots + \boldsymbol{u}_\gamma z^\gamma\\ \boldsymbol{v}(z) &= \boldsymbol{v}_0 + \boldsymbol{v}_1 z + \boldsymbol{v}_2 z^2 + \cdots + \boldsymbol{v}_\gamma z^\gamma\end{aligned}$$

and

$$G(z) = C(zI - A)^{-1}B + D.$$

We say then that the four matrices (A, B, C, D) are the *state-space* representation of the code $\mathcal{C}$. When the matrix D is invertible we say that the code $\mathcal{C}$ is *globally invertible*.

Let us notice that the system (1) drives us to write the sequence $\{\boldsymbol{v}_t\}_{t=0}^{\gamma}$ as

$$\begin{bmatrix}\boldsymbol{v}_0\\ \boldsymbol{v}_1\\ \vdots\\ \boldsymbol{v}_{\gamma-1}\\ \boldsymbol{v}_\gamma\end{bmatrix} =$$

$$\begin{bmatrix} D & & & & \\ CB & D & & & \\ CAB & CB & D & & \\ \vdots & \vdots & \vdots & \ddots & \\ CA^{\gamma-1}B & CA^{\gamma-2}B & \ldots & \ldots & D\end{bmatrix}\begin{bmatrix}\boldsymbol{u}_0\\ \boldsymbol{u}_1\\ \vdots\\ \boldsymbol{u}_\gamma\end{bmatrix}. \qquad (2)$$

3 Description of the cryptosystem

From now on we consider $\mathbb{F} = \mathbb{Z}_2$. Let us suppose that we have $q+1$ pairs convolutional codes of ratio k/k

$$(A_j(b), B_j(b), C_j(b), D_j(b)),$$

not necesarely with the same complexity, such that $D_j(b)$ is invertible for $j = 0, 1, \ldots, q$ and $b \in \{0, 1\}$. In order to encrypt a sequence

$$\boldsymbol{u} = (\boldsymbol{u}_0, \boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_p)$$

where each block $\boldsymbol{u}_i$ for $i = 0, 1, \ldots, p$ has length k, we follow the steps presented below.

Let $j = 0, 1, 2\ldots, q$ and consider

$$\boldsymbol{u}^j = \left(\boldsymbol{u}_0^j, \boldsymbol{u}_1^j, \boldsymbol{u}_2^j, \ldots, \boldsymbol{u}_p^j\right)$$

with $\boldsymbol{u}^0 = \boldsymbol{u}$, i.e., the initial information sequence, and $\boldsymbol{u}^j = \boldsymbol{v}^{j-1}$, i.e., the output of the previous encoder. Let us suppose that

$$b_i^j = w(\boldsymbol{u}_i^j), \text{ for } i = 0, 1, 2, \ldots, p-1 \qquad (3)$$

where $w(\cdot)$ denotes the Hamming weight modulo 2 of a sequence.

Step 0.

We calculate the word

$$\boldsymbol{v}_0^j(0) = D_j(0)\boldsymbol{u}_0^j$$

and the states

$$\boldsymbol{x}_1^j(0) = B_j(0)\boldsymbol{u}_0^j, \quad \boldsymbol{x}_1^j(1) = B_j(1)\boldsymbol{u}_0^j.$$

Step $\boldsymbol{i}$, for $i = 1, 2, \ldots, p$.

Since $w(\boldsymbol{u}_{i-1}^j) = b_{i-1}^j$, we calculate the word

$$\boldsymbol{v}_i^j(b_{i-1}^j) = C_j(b_{i-1}^j)\boldsymbol{x}_i^j(b_{i-1}^j) + D_j(b_{i-1}^j)\boldsymbol{u}_i^j$$

and the states

$$\boldsymbol{x}_{i+1}^j(0) = A_j(0)\boldsymbol{x}_i^j(0) + B_j(0)\boldsymbol{u}_i^j\,,$$

$$\boldsymbol{x}_{i+1}^j(1) = A_j(1)\boldsymbol{x}_i^j(1) + B_j(1)\boldsymbol{u}_i^j\,.$$

In the last step, i.e., for $i = p$, it is not necessary to calculate the states, due to the fact that we are not going to need them.

Thus, we obtain the sequence

$$\begin{aligned}\boldsymbol{v}^j &= \left(\boldsymbol{v}_0^j(b_0^j), \boldsymbol{v}_1^j(b_1^j), \boldsymbol{v}_2^j(b_2^j), \ldots, \boldsymbol{v}_p^j(b_p^j)\right)\\ &= \left(\boldsymbol{v}_0^j, \boldsymbol{v}_1^j, \boldsymbol{v}_2^j, \ldots, \boldsymbol{v}_p^j\right).\end{aligned}$$

The encrypted sequence is, therefore, $\boldsymbol{v} = \boldsymbol{v}^q$, i.e., we receive the sequence

$$\boldsymbol{v} = (\boldsymbol{v}_0, \boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_p)\,.$$

Now, with the encrypted sequence, the systems and the criteria to decide which systems do we use to encrypt, we should be able to recover the initial sequence

$$\boldsymbol{u} = (\boldsymbol{u}_0, \boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_p)\,.$$

Let us see each decryption step. We study separately how to obtain the first block $\boldsymbol{u}_0$, and then the rest of blocks $\boldsymbol{u}_i$ for $i = 1, 2, \ldots, p$.

Obtaining of the block $\boldsymbol{u}_0$ from the block $\boldsymbol{v}_0$.

Step 0.

Considering the way in which we have obtained the sequence $\boldsymbol{v}$, we know that $\boldsymbol{v}_0 = \boldsymbol{v}_0^q = \boldsymbol{v}_0^q(0)$. Now, since

$$\boldsymbol{v}_0^q(0) = D_q(0)\boldsymbol{u}_0^q$$

and $D_q(0)$ is invertible, we are able to calculate $\boldsymbol{u}_0^q$, which in turn allows us to calculate

$$\boldsymbol{x}_1^q(0) = B_q(0)\boldsymbol{u}_0^q, \quad \boldsymbol{x}_1^q(1) = B_q(1)\boldsymbol{u}_0^q.$$

Furthermore, known $\boldsymbol{u}_0^q$, we know that $w(\boldsymbol{u}_0^q) = b_0^q$, hence, in order to calculate $\boldsymbol{v}_1^q$ the system $(A_q(b_0^q), B_q(b_0^q), C_q(b_0^q), D_q(b_0^q))$ was chosen, i.e., $\boldsymbol{v}_1^q = \boldsymbol{v}_1^q(b_0^q)$.

Step $\boldsymbol{j}$, for $j = 1, 2, \ldots, q$.

We know that $\boldsymbol{u}_0^{q-j+1} = \boldsymbol{v}_0^{q-j} = \boldsymbol{v}_0^{q-j}(0)$. Now, since

$$\boldsymbol{v}_0^{q-j}(0) = D_{q-j}(0)\boldsymbol{u}_0^{q-j}$$

and $D_{q-j+1}(0)$ is invertible, we are able to calculate $\boldsymbol{u}_0^{q-j}$, which in turn allows us to calculate

$$\boldsymbol{x}_1^{q-j}(0) = B_{q-j}(0)\boldsymbol{u}_0^{q-j}, \ \boldsymbol{x}_1^{q-j}(1) = B_{q-j}(1)\boldsymbol{u}_0^{q-j}.$$

In addition, known $\boldsymbol{u}_0^{q-j}$, we know that $w(\boldsymbol{u}_0^{q-j}) = b_0^{q-j}$, consequently, in order to calculate $\boldsymbol{v}_1^{q-j}$ the system

$$(A_{q-j}(b_0^{q-j}), B_{q-j}(b_0^{q-j}), C_{q-j}(b_0^{q-j}), D_{q-j}(b_0^{q-j})),$$

was chosen, i.e., $\boldsymbol{v}_1^{q-j} = \boldsymbol{v}_1^{q-j}(b_0^{q-j})$.

Let us notice that, considering the way in which we have defined the cryptosystem, $\boldsymbol{u}_0^0 = \boldsymbol{u}_0$, i.e., we have obtained the block $\boldsymbol{u}_0$.

Obtaining of the block u_i from the block v_i, for $i = 1, 2, \ldots, p$.

Step 0.

Considering the way in which we have constructed the sequence $\boldsymbol{v}$ and the step 0 of the previous block, we know $\boldsymbol{v}_i = \boldsymbol{v}_i^q = \boldsymbol{v}_i^q(b_{i-1}^q)$. Now, since

$$\boldsymbol{v}_i^q(b_{i-1}^q) = C_q(b_{i-1}^q)\boldsymbol{x}_i^q(b_{i-1}^q) + D_q(b_{i-1}^q)\boldsymbol{u}_i^q$$

and $D_q(b_{i-1}^q)$ is invertible, we are able to calculate $\boldsymbol{u}_i^q$, which in turn allows us to calculate

$$\boldsymbol{x}_{i+1}^q(0) = A_q(0)\boldsymbol{x}_i^q(0) + B_q(0)\boldsymbol{u}_i^q \ ,$$

$$\boldsymbol{x}_{i+1}^q(1) = A_q(1)\boldsymbol{x}_i^q(1) + B_q(1)\boldsymbol{u}_i^q \ .$$

Furthermore, known $\boldsymbol{u}_i^q$, we know that $w(\boldsymbol{u}_i^q) = b_i^q$, therefore, in order to calculate $\boldsymbol{v}_{i+1}^q$ the system $(A_q(b_i^q), B_q(b_i^q), C_q(b_i^q), D_q(b_i^q))$ was chosen, i.e., $\boldsymbol{v}_{i+1}^q = \boldsymbol{v}_{i+1}^q(b_i^q)$.

Step $\boldsymbol{j}$, for $j = 1, 2, \ldots, q$.

Considering the way in which the intermediate sequences are obtained and the step j of the previous block, we know that $\boldsymbol{u}_i^{q-j+1} = \boldsymbol{v}_i^{q-j} = \boldsymbol{v}_i^{q-j}(b_{i-1}^{q-j})$. Now, since

$$\boldsymbol{v}_i^{q-j}(b_{i-1}^q)$$
$$= C_{q-j}(b_{i-1}^{q-j})\boldsymbol{x}_i^{q-j}(b_{i-1}^{q-j}) + D_{q-j}(b_{i-1}^{q-j})\boldsymbol{u}_i^{q-j}$$

and $D_{q-j}(b_{i-1}^{q-j})$ is invertible, we are able to calculate $\boldsymbol{u}_i^{q-j}$, which in turn allows us to calculate

$$\begin{aligned}
&\boldsymbol{x}_{i+1}^{q-j+1}(0)\\
&= A_{q-j+1}(0)\boldsymbol{x}_i^{q-j+1}(0) + B_{q-j+1}(0)\boldsymbol{u}_i^{q-j+1},\\
&\boldsymbol{x}_{i+1}^{q-j+1}(1)\\
&= A_{q-j+1}(1)\boldsymbol{x}_i^{q-j+1}(1) + B_{q-j+1}(1)\boldsymbol{u}_i^{q-j+1}.
\end{aligned}$$

In addition, known $\boldsymbol{u}_i^{q-j}$, we know that $w(\boldsymbol{u}_i^{q-j}) = b_i^{q-j}$, hence, in order to calculate $\boldsymbol{v}_{i+1}^{q-j}$ the system

$$(A_{q-j}(b_i^{q-j}), B_{q-j}(b_i^{q-j}), C_{q-j}(b_i^{q-j}), D_{q-j}(b_i^{q-j})),$$

was chosen, i.e., $\boldsymbol{v}_{i+1}^{q-j} = \boldsymbol{v}_{i+1}^{q-j}(b_i^{q-j})$.

Let us notice that, considering the way in which we have defined the cryptosystem, $\boldsymbol{u}_i^0 = \boldsymbol{u}_i$, i.e., we have obtained the block $\boldsymbol{u}_i$.

Thus, following this process we recover the initial sequence

$$\boldsymbol{u} = (\boldsymbol{u}_0, \boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_p) \, .$$

4 Security of the cryptosystem

Let us assume that the cryptanalyst knows the encrypted sequence $\boldsymbol{v}$. Since he knows the technique used to encrypt, but he does not know neither the number of systems, nor the matrices, nor the size of the matrices, the only thing he knows is that the sequence $\boldsymbol{x}$, corresponding to the plaintext, satisfies an equation with the form

$$\boldsymbol{v} = M\boldsymbol{x}$$

where M is, in this case, an invertible matrix. Here, the cryptanalyst does not know neither $\boldsymbol{x}$ nor M. Though he knows that M is, as said in equation (2), lower block triangular, he does not know how many blocks it has, neither the size of them. Consequently, an exhaustive key search seems to be unfeasible as long as the size of the cryptosystem involved matrices is considerable.

Let us suppose that the cryptanalyst knows some pairs

$$(\boldsymbol{x}_l, \boldsymbol{v}_l), \quad \text{for} \quad l = 1, 2, \ldots, r$$

where $\boldsymbol{v}_l$ is the encrypted sequence corresponding to the plaintext $\boldsymbol{x}_l$.

Let us notice, first of all, that the matrix M is not the same for every sequence. As we choose the intermediate systems depending on the sequence we are encrypting at each time, we will have as many matrices as plaintext sequences.

Therefore, the cryptanalyst knows

$$\boldsymbol{v}_l = M_l \boldsymbol{x}_l, \quad \text{for} \quad l = 1, 2, \ldots, r,$$

with M_l an invertible lower block triangular matrix. As in the previous case, the cryptanalyst does not know neither the size of the blocks nor the number of blocks, hence, the attack also seems to be nonviable.

5 Conclusions

In this article we have presented a concatenation of convolutinal codes based symmetric cryptosystem. We have described the encryption and decryption processes and have analyzed its security concluding that an exhaustive key search and a known-plaintext attack are not feasible for high values of the parameters.

References:

[1] A. DHOLAKIA. *Introduction to Convolutional Codes with Applications*. Kluwer Academic Publishers, Boston, MA, 1994.

[2] G. D. FORNEY, JR., R. JOHANNESSON and Z.-X. WAN. Minimal and canonical rational generator matrices for convolutional codes. *IEEE Transactions on Information Theory*, **42(6)**: 1865–1880 (1993).

[3] R. JOHANNESSON and K. S. ZIGANGIROV. *Fundamentals of Convolutional Coding*. IEEE Press, New York, NY, 1999.

[4] R. LOBO, D. BITZER and M. VOUK. Locally invertible m-dimensional convolutional codes. Technical Report TR-2004-33, Department of Computer Science, North Carolina State University, Raleigh, NC 27695, 2004.

[5] R. J. MCELIECE. A public-key cryptosystem based on algebraic coding theory. DNS Progress Report 42–44, Jet Propulsion Laboratory, 1978.

[6] R. J. MCELIECE. The algebraic theory of convolutional codes. In V. S. PLESS and W. C. HUFFMAN (editors), *Handbook of Coding Theory*, pages 1065–1138. Elsevier, North-Holland, 1998.

[7] A. J. MENEZES, P. C. VAN OORSCHOT and S. A. VANSTONE. *Handbook of Applied Cryptography*. CRC Press, Boca Raton, FL, 1996.

[8] J. ROSENTHAL. Connections between linear systems and convolutional codes. In B. MARCUS and J. ROSENTHAL (editors), *Codes, Systems and Graphical Models*, volume 123 of *IMA*, pages 39–66. Springer-Verlag, Berlin, 2001.

[9] J. ROSENTHAL, J. SCHUMACHER and E. V. YORK. On behaviors and convolutional codes. *IEEE Transactions on Information Theory*, **42(6)**: 1881–1891 (1996).

[10] B. SCHNEIER. *Applied Cryptography*. John Wiley & Sons, New York, NY, second edition, 1996.

[11] E. V. YORK. *Algebraic Description and Construction of Error Correcting Codes: A Linear Systems Point of View*. PhD Thesis, Department of Mathematics, University of Notre Dame, Indiana, USA, May 1997.