МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

UDC 519.7 DOI 10.17223/20710410/40/3

ASYMMETRIC CRYPTOSYSTEMS ON BOOLEAN FUNCTIONS[1]

G. P. Agibalov, I. A. Pankratova

National Research Tomsk State University, Tomsk, Russia

Here, we define an asymmetric substitution cryptosystem combining both a public key cipher and a signature scheme with the functional keys. A public key in the cryptosystem is a vector Boolean function $f(x_1,\ldots,x_n)$ of a dimension n. This function is obtained by permutation and negation operations on variables and coordinate functions of a bijective vector Boolean function $g(x_1,\ldots,x_n) = (g_1(x_1,\ldots,x_n),\ldots,g_n(x_1,\ldots,x_n))$. The function g is called a generating function of the cryptosystem. For each $i\in\{1,\ldots,n\}$, its coordinate function $g_i(x_1,\ldots,x_n)$ is assumed to be specified in a constructive way and to have a polynomial (in n) complexity. A private key of the cryptosystem is the function f^{-1}, that is, the inverse of f. The existence of f^{-1} follows from the bijectiveness of g and preserving this property by permutation and negation operations. Function g and its coordinates $g_1,\ldots,g_n$ are public parameters of the cryptosystem. (A variant of the cryptosystem allows to include them into the private key). Of course, the permutation and negation operations by which a public key is computed from the generating function must be secret as private exponents in RSA and ElGamal cryptosystems. A block P of a plaintext is encrypted to a block C of a ciphertext by the rule $C = f(P)$, and C is decrypted to P by the rule $P = f^{-1}(C)$. A signature on a message M is computed as $S = f^{-1}(P)$, and its validation is proved by verifying the equality $M = f(S)$. This cryptosystem is believed to resist classical and quantum computers attacks. Its security is based on the difficulty of inverting large bijective vector Boolean functions. Cryptanalysis of the cryptosystem shows that its computational complexity can reach the value $\mathrm{O}(n!2^n)$.

Keywords: *vector Boolean functions, invertibility, asymmetric substitution cryptosystem, cryptanalysis.*

Introduction

Public-key cryptosystems are usually constructed on the base of number theory or algebraic structures and are very susceptible to quantum attacks. Perhaps the only exception to this rule are finite automaton public key cryptosystems [1]. In this paper, we suggest a public-key cryptosystem based on an invertible system of n Boolean functions which is variable like a cryptographic key by the permutation and negation operations on system's arguments and coordinates. We call it ACBF — Asymmetric Cryptosystem on Boolean Functions. The cryptosystem typically consists of two parts — a public-key cipher and a signature scheme. A general cryptanalysis scheme is described for both of them. According to this scheme, some particular known playntext (and known message) attacks are proposed for the universal ACBF and for its derivatives with some permutation and negation operations being identities. Estimates for computational complexity of these

[1] The authors were supported by the RFBR-grant no. 17-01-00354.

attacks are given too. The most of them is $\mathrm{O}(n!2^n)$. For each of fifteen ACBF we considered, the proposed attacks on its cipher and signature scheme happened to have the same estimate of computational complexity.

1. Definition

In [2], we have defined a symmetric block substitution cipher with the functional keys. Here, by using the same construction, we define an asymmetric substitution cryptosystem including both a public key cipher and a signature scheme with the functional keys. To give a formal definition of this cryptosystem, we first define the permutation and negation operations. Let n be an integer, $n \geqslant 2$, and $\mathbb{S}_n$ be the set of all permutations of the row $(1\,2\,\ldots\,n)$, that is, $\mathbb{S}_n = \{(i_1 i_2 \ldots i_n) : i_j \in \{1,2,\ldots,n\}, j \neq r \Rightarrow i_j \neq i_r; j,r \in \{1,\ldots,n\}\}$. A permutation $\pi = (i_1 i_2 \ldots i_n) \in \mathbb{S}_n$ is called a *permutation operation* if the result of its application to any word $w = w_1 w_2 \ldots w_n$ is the word $\pi(w) = w_{i_1} w_{i_2} \ldots w_{i_n}$. A Boolean vector $\sigma = b_1 b_2 \ldots b_n \in \mathbb{F}_2^n$ is called a *negation operation* if the result of its application to a string $\alpha = a_1 a_2 \ldots a_n$ of Boolean values (constants, variables or functions) $a_1,\ldots,a_n$ is the string $\alpha^\sigma = a_1^{b_1} a_2^{b_2} \ldots a_n^{b_n}$ where for a and b in $\mathbb{F}_2$, $a^b = a$ if $b = 1$ and $a^b = \neg a$ if $b = 0$. The permutation and negation operations π and σ are called *identity* and denoted by 1 if $\pi = (1\,2\,\ldots\,n)$ and $\sigma = 11\ldots1$ respectively.

Formally, our *asymmetric cryptosystem on Boolean functions* is a three-tuple $\mathcal{C} = (X, K, Y)$ where X is the set of plaintexts, or messages, $X \subseteq \mathbb{F}_2^n$, Y is the set of ciphertexts or signatures, $Y \subseteq \mathbb{F}_2^n$, and $K = K_1 \times K_2$ is the set of keys, K_1 — the set of public keys, $K_1 \subseteq K_n(g) = \{f(x) : f(x) = \pi_2(g^{\sigma_2}(\pi_1(x^{\sigma_1}))); \sigma_1,\sigma_2 \in \mathbb{F}_2^n; \pi_1,\pi_2 \in \mathbb{S}_n\}$; $x = (x_1,\ldots,x_n)$ is a string of different Boolean variables, $g : \mathbb{F}_2^n \to \mathbb{F}_2^n$ is a bijective vector Boolean function $g(x) = g_1(x) g_2(x) \ldots g_n(x)$ (we call it *generating function* of $\mathcal{C}$) with all its coordinate functions $g_1(x),\ldots,g_n(x)$ specified in a constructive way and computed with a polynomial (in n) time complexity; π_1,π_2 and σ_1,σ_2 are, respectively, permutation and negation operations (we call them *key parameters* of $\mathcal{C}$); and $K_2 = \{f^{-1} : f \in K_1\}$ — the set of private keys. In the case of $X = Y = \mathbb{F}_2^n$ and $K_1 = K_n(g)$, we call $\mathcal{C}$ a *universal* ACBF.

In $\mathcal{C}$, as in any asymmetric cipher, a public key f is used to encrypt a plaintext x and the private key f^{-1} — to decrypt the corresponding ciphertext y, namely: $y = f(x)$ and $x = f^{-1}(y)$ for $x \in X$, $y \in Y$, $f \in K_1$, $f^{-1} \in K_2$. Also, in $\mathcal{C}$, as in any digital signature scheme with appendix (the signed message), a private key f^{-1} is used to sign a message x and the public key f — to verify signatures, namely: the signature for a message x is $s = f^{-1}(x)$ and a signature s on a message x is valid iff $f(s) = x$.

To provide the necessary property of ease (polynomial time complexity) of computing the functions f and f^{-1}, the generating function g itself and its inverse g^{-1} should have this property too. In this case, the values $y = f(x) = \pi_2(g^{\sigma_2}(\pi_1(x^{\sigma_1})))$ and $x = f^{-1}(y) = [\pi_1^{-1}(g^{-1}((\pi_2^{-1}(y))^{\sigma_2}))]^{\sigma_1}$ would be computed with a polynomial complexity. The polynomial computational complexity of each coordinate in generating function g guarantees a polynomial complexity of computing g itself. This is true if, for example, every coordinate function $g_i(x)$ essentially depends on some $s_i \leqslant s_0$ variables $x_{i_1},\ldots,x_{i_{s_i}}$ from the string x, that is, $g_i(x) = h_i(x_{i_1},\ldots,x_{i_{s_i}})$ for a function $h_i : \mathbb{F}_2^{s_i} \to \mathbb{F}_2$ and s_0 is a small enough integer, $1 \leqslant s_0 \leqslant n$. As for providing a polynomial complexity of computing the function g^{-1}, there are many ways to choose g in $\mathcal{C}$ preserving its polynomial complexity in g^{-1}. One of them is the following: $g(x) = g^{(1)}(x) \ldots g^{(r)}(x)$, $1 \leqslant r \leqslant n$, $g^{(i)}(x) = g_{i_1}(x) \ldots g_{i_{s_i}}(x) = h_{i_1}(x^{(i)}) \ldots h_{i_{s_i}}(x^{(i)}) = h^{(i)}(x^{(i)})$, $x^{(i)} = x_{i_1} \ldots x_{i_{s_i}}$, $h^{(i)} : \mathbb{F}_2^{s_i} \to \mathbb{F}_2^{s_i}$ is a bijection, $s_1 + s_2 + \ldots + s_r = n$, $i \neq j \Rightarrow \{i_1,\ldots,i_{s_i}\} \cap \{j_1,\ldots,j_{s_j}\} = \varnothing$, $i,j \in \{1,2,\ldots,r\}$. In this case, $g(x) = h^{(1)}(x^{(1)}) \ldots h^{(r)}(x^{(r)})$ and if $y = g(x)$, then

$y^{(i)} = y_{i_1} \dots y_{i_{s_i}} = h^{(i)}(x^{(i)})$, $h^{(i)^{-1}}(y^{(i)}) = x^{(i)}$, $g^{-1}(y) = h^{(1)^{-1}}(y^{(1)}) \dots h^{(r)^{-1}}(y^{(r)})$. That is, $g^{-1}(y)$ is computed with a polynomial complexity.

The security of ACBF $\mathcal{C}$ is based on the difficulty of inverting large bijective vector Boolean functions, that is, of computing $x = f^{-1}(y)$ for $y = f(x)$. For an opponent or cryptanalyst, who (this is believed) doesn't know the values of key parameters π_1, π_2, σ_1 and σ_2 in f, this problem is really difficult one with an exponential time complexity of decision algorithm.

2. Cryptanalysis problem

The cryptanalysis problem that we study for ACBF $\mathcal{C}$ in the paper is the secret key recovery, assuming some plaintexts or messages and the corresponding ciphertexts or signatures are known. If $\mathcal{C}$ is a cipher, the problem is set as follows: given $f(x) \in K_1$, $P_l \in X$, and $C_l = f(P_l)$, $l = 1, 2, \dots, m$, compute $f^{-1}(y)$. Otherwise, that is, if $\mathcal{C}$ is a signature scheme, the problem looks like the following: given $f(x) \in K_1$, $M_l \in X$, and $S_l = f^{-1}(M_l)$, $l = 1, 2, \dots, m$, compute $f^{-1}(x)$. Further, the problem in the first case is called the cipher cryptanalysis, in the second case — the signature cryptanalysis. Any methods solving them are called attacks on cipher and on signature scheme respectively. Aiming to recover the secret key, they are total break attacks. Besides, when we say that the public key $f(x)$ is given, we suppose that everybody has a possibility to compute its value at any point x for a polynomial time, but no cryptanalyst (opponent) knows the parameters $\pi_1, \pi_2, \sigma_1, \sigma_2$ of $f(x)$ (compare, for example, with g^a over $\mathbb{Z}_p$ in ElGamal cipher).

Below we describe some attacks on ciphers and on signature schemes of a universal ACBF $\mathcal{C}$ and of its particular derivatives which are ACBF obtained from $\mathcal{C}$ by replacing some key parameters with the identity operation 1. Here is a more correct definition of this concept. Let $I = \{\pi_1, \pi_2, \sigma_1, \sigma_2\}$, $J \subset I$, and $\mathcal{C}(J) = (X, K(J), Y)$ where $X = Y = \mathbb{F}_2^n$, $K(J) = K_1(J) \times K_2(J)$, $K_2(J) = \{f^{-1}(x) : f(x) \in K_1(J)\}$, $K_1(J) = K_n(g, J)$, and $K_n(g, J)$ is the set of all functions $f(x) = \pi_2(g^{\sigma_2}(\pi_1(x^{\sigma_1})))$ such that $\pi_1, \pi_2 \in \mathbb{S}_n$, $\sigma_1, \sigma_2 \in \mathbb{F}_2^n$, and each key parameter from $I \setminus J$ is the identity. By the definition, $K_n(g, J) \subseteq K_n(g)$, therefore $\mathcal{C}(J)$ is really an ACBF; $K_n(g, I) = K_n(g)$, therefore $\mathcal{C}(I) = \mathcal{C}$; $K_n(g, \varnothing) = \{g(x)\}$, therefore $\mathcal{C}(\varnothing)$ is a trivial cryptosystem. We call $\mathcal{C}(J)$ a *particular derivative* of the $\mathcal{C}$ if $\varnothing \neq J \neq I$. So, for any universal ACBF $\mathcal{C}$, we have 14 particular derivatives $\mathcal{C}(J)$ in total.

Note that for any vector-columns a, σ in $\mathbb{F}_2^n$ and a permutation $\pi = (i_1 i_2 \dots i_n) \in \mathbb{S}_n$, if $c = \neg\sigma$, $\mathrm{w}(a)$ is the weight of a, that is the number of 1's in a, and $T = (t_{kj})$ is a permutation matrix of order n over $\mathbb{F}_2$ where $t_{kj} = 1 \Leftrightarrow j = i_k$ for all $k, j \in \{1, 2, \dots, n\}$, then $a^\sigma = a \oplus c$, $\pi(a) = Ta$, and $\mathrm{w}(Ta) = \mathrm{w}(a)$. Further, we use these facts without additional explanations and call T the matrix of the permutation π. Moreover, we introduce the following notation: A and D are the matrices of permutations π_1 and π_2 respectively and b and d are the vector-columns $\neg\sigma_1$ and $\neg\sigma_2$ respectively. This allows us to apply the symbols of variables A, D, b, d instead of symbols of operations $\pi_1, \pi_2, \sigma_1, \sigma_2$ respectively in the sets I, J as well as in the formulas for $f(x), f^{-1}(x)$ and so on. The fact of such replacement is denoted by the sign $\simeq$. For example, $\{\pi_1, \sigma_2\} \simeq \{A, d\}$.

3. General scheme of attack

Here is the general scheme of an attack on the cipher in an ACBF $\mathcal{C}(J)$.

1. Express the function $f^{-1}(y)$ by a formula in the set of variables and operations

$$J \cup \{y, g, \oplus, \cdot, ^{-1}\}.$$

2. Record the system of equations E expressing the dependence of variables from J on the values P_l, C_l, $1 \leqslant l \leqslant n$, by means of operations $\oplus, \cdot, ^{-1}$ and function g.
3. Solve the system E in unknowns from J using a proper method [3].
4. Substitute the variables from J in formula for $f^{-1}(y)$ by their values from the solution of the system E. The resulting formula is the result of the attack.
5. Estimate the computational complexity of the attack as a time complexity of solving the system of equations E.

The description of the general scheme of an attack on the signature scheme in an ACBF $\mathcal{C}(J)$ is obtained from this scheme for ciphers by substitution $f^{-1}(x)$, M_l, and S_l for $f^{-1}(y)$, C_l, and P_l respectively. The attacks, described below, on universal ACBF $\mathcal{C} = \mathcal{C}(I)$ and on its particular derivatives $\mathcal{C}(J)$ are constructed according to the general scheme. In the case of nonlinear equations in the system E, this system in them is solved by the method of linearization set [3, 4] (further we call it briefly method of LS). The vector weight invariance related to multiplying by a permutation matrix is used to decrease the computational complexity of some of these attacks in practice.

4. Attacks on universal ACBF

4.1. Attack on cipher

We have $y = f(x) = \pi_2(g^{\sigma_2}(\pi_1(x^{\sigma_1}))) = D(g(A(x \oplus b)) \oplus d)$; $D^{-1}y \oplus d = g(A(x \oplus b))$, $A^{-1}(g^{-1}(D^{-1}y \oplus d)) \oplus b = x$, $f^{-1}(y) = x$, and $C_l = f(P_l)$, $l = 1, 2, \ldots, m$. Hence,

$$f^{-1}(y) = A^{-1}g^{-1}(D^{-1}y \oplus d) \oplus b$$

where (D^{-1}, d, b, A) is a solution of the system of equations

$$D^{-1}C_l \oplus d = g(A(P_l \oplus b)), \quad l = 1, 2, \ldots, m,$$

which is solved by the method of LS, namely by assigning in turn the different values to the variables A, b and solving the resulting system of linear equations with unknowns in D^{-1} and d. So, the computational complexity of the attack is $\mathrm{O}(n!2^n)$.

4.2. Attack on signature scheme

We have $S_l = f^{-1}(M_l) = A^{-1}g^{-1}(D^{-1}M_l \oplus d) \oplus b$, $l = 1, 2, \ldots, m$, and

$$f^{-1}(x) = A^{-1}g^{-1}(D^{-1}x \oplus d) \oplus b$$

where (D^{-1}, d, b, A) is a solution of the system of equations

$$D^{-1}M_l \oplus d = g(A(S_l \oplus b)), \quad l = 1, 2, \ldots, m,$$

which is solved by the method of LS, that is, by assigning in turn the different values to the variables A, b and solving the resulting system of linear equations with unknowns D^{-1} and d. So, the computational complexity of the attack is $\mathrm{O}(n!2^n)$.

5. Attacks on particular derivatives of universal ACBF

5.1. Attacks on ciphers

Given $J \subset \{\pi_1, \pi_2, \sigma_1, \sigma_2\} \simeq \{A, D, b, d\}$, $f(x) \in K_n(g, J)$, $P_l \in \mathbb{F}_2^n$, and $C_l = f(P_l)$, $l = 1, 2, \ldots, m$, compute $f^{-1}(y)$. Consider the possible cases.

1. $J = \{\sigma_1\} \simeq \{b\}$.

We have $y = f(x) = g(x^{\sigma_1}) = g(x \oplus b)$ and $b = \neg\sigma_1$, therefore $g^{-1}(y) = x \oplus b$ and $f^{-1}(y) = x$. Hence,

$$f^{-1}(y) = g^{-1}(y) \oplus b$$

where

$$b = P_l \oplus g^{-1}(C_l), \quad l = 1, 2, \ldots, m.$$

So, $b = P_1 \oplus g^{-1}(C_1)$. Computational complexity of this attack is a polynomial in n.

2. $J = \{\pi_1\} \simeq \{A\}$.

Here, $y = f(x) = g(\pi_1(x)) = g(Ax)$ and A is the matrix of π_1; $Ax = g^{-1}(y)$, $x = A^{-1}g^{-1}(y)$, $f^{-1}(y) = x$. Hence,

$$f^{-1}(y) = A^{-1}g^{-1}(y),$$

and the matrix A is a solution of the system of linear equations

$$AP_l = g^{-1}(C_l), \quad l = 1, 2, \ldots, m,$$

which is solved for a polynomial time.

In case $m = 1$, this system has $r = w!(n - w)!$ of solutions where $w = \mathrm{w}(P_1)$ and r is the product of permutation quantities for ones and for zeros in P_1. The computational complexity of this attack coincides the time complexity of solving the system of linear equations problem and does not exceed a polynomial in n.

3. $J = \{\pi_1, \sigma_1\} \simeq \{A, b\}$.

In this case, $y = f(x) = g(\pi_1(x^{\sigma_1})) = g(A(x \oplus b))$; $g^{-1}(y) = A(x \oplus b)$, $\mathrm{w}(x \oplus b) = \mathrm{w}(g^{-1}(y))$, $A^{-1}g^{-1}(y) \oplus b = x$, $f^{-1}(y) = x$.

Define $B_l \subseteq \mathbb{F}_2^n$ by induction on $l = 1, 2, \ldots, m$, namely let $B_1 = \{\underline{b} : \mathrm{w}(P_1 \oplus \underline{b}) = \mathrm{w}(g^{-1}(C_1))\}$ and $B_l = \{\underline{b} : \underline{b} \in B_{l-1}, \mathrm{w}(P_l \oplus \underline{b}) = \mathrm{w}(g^{-1}(C_l))\}$, $2 \leqslant l \leqslant m$. Then

$$f^{-1}(y) = A^{-1}g^{-1}(y) \oplus b$$

where (A, b) is a solution of the following system of equations

$$A(P_l \oplus b) = g^{-1}(C_l), \quad l = 1, 2, \ldots, m,$$

which is solved, using the method of LS, by assigning in turn the different values from the set B_m to the variable b and solving the resulting system of linear equations with unknowns in A. The computational complexity of the attack is $\mathrm{O}(\mathrm{C}_n^{n/2})$.

4. $J = \{\sigma_2\} \simeq \{d\}$.

$y = f(x) = g^{\sigma_2}(x) = g(x) \oplus d$ and $d = \neg\sigma_2$; $g^{-1}(y \oplus d) = x$, $f^{-1}(y) = x$. Hence,

$$f^{-1}(y) = g^{-1}(y \oplus d)$$

where

$$d = C_l \oplus g(P_l), \quad l = 1, 2, \ldots, m.$$

So, $d = C_1 \oplus g(P_1)$. Computational complexity of the attack is a polynomial in n.

5. $J = \{\sigma_1, \sigma_2\} \simeq \{b, d\}$.

$y = f(x) = g^{\sigma_2}(x^{\sigma_1}) = g(x \oplus b) \oplus d$ where $b = \neg\sigma_1$, $d = \neg\sigma_2$; $x \oplus b = g^{-1}(y \oplus d)$, $g^{-1}(y \oplus d) \oplus b = x$, $f^{-1}(y) = x$. Hence,

$$f^{-1}(y) = g^{-1}(y \oplus d) \oplus b$$

where (b, d) is a solution of the following system of equations

$$g(P_l \oplus b) \oplus d = C_l, \quad l = 1, 2, \ldots, m,$$

and can be computed by the method of LS, that is, by assigning in turn the different values to b and solving the resulting system of linear equations with unknowns in d. The complexity of this attack is $\mathrm{O}(2^n)$.

6. $J = \{\pi_1, \sigma_2\} \simeq \{A, d\}$.

$y = f(x) = g^{\sigma_2}(\pi_1(x)) = g(Ax) \oplus d$; $Ax = g^{-1}(y \oplus d)$, $\mathrm{w}(x) = \mathrm{w}(g^{-1}(y \oplus d))$, $A^{-1}g^{-1}(y \oplus \oplus d) = x$, $f^{-1}(y) = x$.

Define $D_l \subseteq \mathbb{F}_2^n$ by induction on $l = 1, 2, \ldots, m$, namely let $D_1 = \{\underline{d} : \mathrm{w}(P_1) = = \mathrm{w}(g^{-1}(C_1 \oplus \underline{d}))\}$ and $D_l = \{\underline{d} : \underline{d} \in D_{l-1}, \mathrm{w}(P_l) = \mathrm{w}(g^{-1}(C_l \oplus \underline{d}))\}$, $2 \leqslant l \leqslant m$. Then

$$f^{-1}(y) = A^{-1}g^{-1}(y \oplus d)$$

where (A, d) is a solution of the following system of equations

$$AP_l = g^{-1}(C_l \oplus d), \quad l = 1, 2, \ldots, m.$$

This system is solved, using the method of LS, by assigning in turn the different values from the set D_m to the variable d and solving the resulting system of linear equations in unknowns in A. The computational complexity of the attack is $\mathrm{O}(\mathrm{C}_n^{n/2})$.

7. $J = \{\pi_1, \sigma_1, \sigma_2\} \simeq \{A, b, d\}$.

$y = f(x) = g^{\sigma_2}(\pi_1(x^{\sigma_1})) = g(A(x \oplus b)) \oplus d$; $A(x \oplus b) = g^{-1}(y \oplus d)$, $\mathrm{w}(x \oplus b) = \mathrm{w}(g^{-1}(y \oplus d))$, $A^{-1}g^{-1}(y \oplus d) = x \oplus b$, $f^{-1}(y) = x$.

For each $\underline{d} \in \mathbb{F}_2^n$, define $B_l(\underline{d}) \subseteq \mathbb{F}_2^n$ by induction on $l = 1, 2, \ldots, m$, namely let $B_1(\underline{d}) = = \{\underline{b} : \mathrm{w}(P_1 \oplus \underline{b}) = \mathrm{w}(g^{-1}(C_1 \oplus \underline{d}))\}$ and $B_l(\underline{d}) = \{\underline{b} : \underline{b} \in B_{l-1}(\underline{d}), \mathrm{w}(P_l \oplus \underline{b}) = \mathrm{w}(g^{-1}(C_l \oplus \underline{d}))\}$, $2 \leqslant l \leqslant m$. Then

$$f^{-1}(y) = A^{-1}g^{-1}(y \oplus d) \oplus b$$

where (A, b, d) is a solution of the following system of equations

$$A(P_l \oplus b) = g^{-1}(C_l \oplus d), \quad l = 1, 2, \ldots, m.$$

This system is solved, using the method of LS, by assigning in turn the different values $(\underline{b}, \underline{d})$ with $\underline{d}$ from $\mathbb{F}_2^n$ and $\underline{b}$ from $B_m(\underline{d})$ to the pair of variables (b, d) and solving the resulting system of linear equations in unknowns in A. The computational complexity of the attack is $\mathrm{O}(2^n \mathrm{C}_n^{n/2})$.

8. $J = \{\pi_2\} \simeq \{D\}$.

$y = f(x) = \pi_2(g(x)) = Dg(x)$ where D is the matrix of the permutation π_2; $D^{-1}y = = g(x)$, $g^{-1}(D^{-1}y) = x$, $f^{-1}(y) = x$. Hence,

$$f^{-1}(y) = g^{-1}(D^{-1}y)$$

where D^{-1} is a solution of the system of linear equations

$$D^{-1}C_l = g(P_l), \quad l = 1, 2, \ldots, m.$$

Computational complexity of this attack is a polynomial in n.

9. $J = \{\pi_2, \sigma_1\} \simeq \{D, b\}$.

$y = f(x) = \pi_2(g(x^{\sigma_1})) = Dg(x \oplus b)$; $D^{-1}y = g(x \oplus b)$, $\mathrm{w}(y) = \mathrm{w}(g(x \oplus b))$, $g^{-1}(D^{-1}y) \oplus \oplus b = x$, $f^{-1}(y) = x$.

Define $B_l \subseteq \mathbb{F}_2^n$ by induction on $l = 1, 2, \ldots, m$, namely let $B_1 = \{\underline{b} : \mathrm{w}(C_1) = \mathrm{w}(g(P_1 \oplus \oplus \underline{b}))\}$ and $B_l = \{\underline{b} : \underline{b} \in B_{l-1}, \mathrm{w}(C_l) = \mathrm{w}(g(P_l \oplus \underline{b}))\}$, $2 \leqslant l \leqslant m$. Then

$$f^{-1}(y) = g^{-1}(D^{-1}y) \oplus b$$

where (D^{-1}, b) is a solution of the system of equations

$$D^{-1}C_l = g(P_l \oplus b), \quad l = 1, 2, \ldots, m,$$

which is solved, using the method of LS, by assigning in turn the different values from B_m to the variable b and solving the resulting system of linear equations with unknowns in D^{-1}. So, the computational complexity of the attack is $\mathrm{O}(\mathrm{C}_n^{n/2})$.

10. $J = \{\pi_1, \pi_2\} \simeq \{A, D\}$.

$y = f(x) = \pi_2(g(\pi_1(x))) = Dg(Ax)$; $D^{-1}y = g(Ax)$, $A^{-1}g^{-1}(D^{-1}y) = x$, $f^{-1}(y) = x$. Hence,

$$f^{-1}(y) = A^{-1}g^{-1}(D^{-1}y)$$

where (A, D^{-1}) is a solution of the system of equations

$$D^{-1}C_l = g(AP_l), \quad l = 1, 2, \ldots, m,$$

which is solved by the method of LS, that is, by assigning in turn the different values to the variable A and solving the resulting system of linear equations with unknowns in D^{-1}. So, the computational complexity of the attack is $\mathrm{O}(n!)$.

11. $J = \{\pi_1, \pi_2, \sigma_1\} \simeq \{A, D, b\}$.

$y = f(x) = \pi_2(g(\pi_1(x^{\sigma_1}))) = Dg(A(x \oplus b))$; $D^{-1}y = g(A(x \oplus b))$, $A^{-1}g^{-1}(D^{-1}y) = x \oplus b$, $f^{-1}(y) = x$. Hence,

$$f^{-1}(y) = A^{-1}g^{-1}(D^{-1}y) \oplus b$$

where (A, D^{-1}, b) is a solution of the system of equations

$$A^{-1}g^{-1}(D^{-1}C_l) \oplus b = P_l, \quad l = 1, 2, \ldots, m,$$

which is solved by the method of LS, that is, by assigning in turn the different values to the variable D^{-1} and solving the resulting system of linear equations with unknowns in A, b. The computational complexity of the attack is $\mathrm{O}(n!)$.

12. $J = \{\pi_2, \sigma_2\} \simeq \{D, d\}$.

$y = f(x) = \pi_2(g^{\sigma_2}(x)) = D(g(x) \oplus d)$; $D^{-1}y = g(x) \oplus d$, $g^{-1}(D^{-1}y \oplus d) = x$, $f^{-1}(y) = x$. Hence,

$$f^{-1}(y) = g^{-1}(D^{-1}y \oplus d)$$

where (D^{-1}, d) is a solution of the system of linear equations

$$D^{-1}C_l \oplus d = g(P_l), \quad l = 1, 2, \ldots, m,$$

which is solved with a polynomial complexity.

13. $J = \{\pi_2, \sigma_1, \sigma_2\} \simeq \{D, b, d\}$.

$y = f(x) = \pi_2(g^{\sigma_2}(x^{\sigma_1})) = D(g(x \oplus b) \oplus d)$; $D^{-1}y = g(x \oplus b) \oplus d$, $g^{-1}(D^{-1}y \oplus d) = x \oplus b$, $f^{-1}(y) = x$. Hence,

$$f^{-1}(y) = g^{-1}(D^{-1}y \oplus d) \oplus b$$

where (D^{-1}, b, d) is a solution of the system of equations

$$D^{-1}C_l \oplus d = g(P_l \oplus b), \quad l = 1, 2, \ldots, m,$$

which is solved by the method of LS, that is, by assigning in turn the different values to the variable b and solving the resulting system of linear equations with unknowns in D^{-1} and d. So, the computational complexity of the attack is $\mathrm{O}(2^n)$.

14. $J = \{\pi_1, \pi_2, \sigma_2\} \simeq \{A, D, d\}$.

$y = f(x) = \pi_2(g^{\sigma_2}(\pi_1(x))) = D(g(Ax) \oplus d)$; $D^{-1}y = g(Ax) \oplus d$, $A^{-1}g^{-1}(D^{-1}y \oplus d) = x$, $f^{-1}(y) = x$. Hence,

$$f^{-1}(y) = A^{-1}g^{-1}(D^{-1}y \oplus d)$$

where (D^{-1}, d, A) is a solution of the system of equations

$$D^{-1}C_l \oplus d = g(AP_l), \quad l = 1, 2, \ldots, m,$$

which is solved by the method of LS, that is, by assigning in turn the different values to the variable A and solving the resulting system of linear equations with unknowns in D^{-1} and d. So, the computational complexity of the attack is $\mathrm{O}(n!)$.

5.2. Attacks on signature schemes

Given $J \subset \{\pi_1, \pi_2, \sigma_1, \sigma_2\} \simeq \{A, D, b, d\}$, $f(x) \in K_n(g, j)$, $M_l \in \mathbb{F}_2^n$, and $S_l = f^{-1}(M_l)$, $l = 1, 2, \ldots, m$, compute $f^{-1}(x)$. Consider the possible cases. As mentioned above, in every case the attack on a signature scheme differs from the attack on a cipher just given by only using variables x, M_l, and S_l instead of y, C_l, and P_l respectively. Taking into account that the attacks on the signature schemes have the great and distinctive significance for cryptography, we describe them without abbreviations.

1. $J = \{\sigma_1\} \simeq \{b\}$.

In this case, $S_l = g^{-1}(M_l) \oplus b$, $l = 1, 2, \ldots, m$, and

$$f^{-1}(x) = g^{-1}(x) \oplus b$$

where

$$b = S_l \oplus g^{-1}(M_l), \quad l = 1, 2, \ldots, m.$$

The computational complexity of the attack is a polynomial in n.

2. $J = \{\pi_1\} \simeq \{A\}$.

$S_l = A^{-1}g^{-1}(M_l)$, $l = 1, 2, \ldots, m$, and

$$f^{-1}(x) = A^{-1}g^{-1}(x)$$

where A is a solution of the system of linear equations

$$AS_l = g^{-1}(M_l), \quad l = 1, 2, \ldots, m,$$

which is solved for a polynomial time.

3. $J = \{\pi_1, \sigma_1\} \simeq \{A, b\}$.

$S_l = A^{-1}g^{-1}(M_l) \oplus b$, $A(S_l \oplus b) = g^{-1}(M_l)$, $\mathrm{w}(S_l \oplus b) = \mathrm{w}(g^{-1}(M_l))$, $l = 1, 2, \ldots, m$, $B_1 = \{\underline{b} : \mathrm{w}(S_1 \oplus \underline{b}) = \mathrm{w}(g^{-1}(M_1))\}$, and $B_l = \{\underline{b} : \underline{b} \in B_{l-1}, \mathrm{w}(S_l \oplus \underline{b}) = \mathrm{w}(g^{-1}(M_l))\}$, $2 \leqslant l \leqslant m$. So,

$$f^{-1}(x) = A^{-1}g^{-1}(x) \oplus b$$

where (A,b) is a solution of the system of equations

$$A(S_l \oplus b) = g^{-1}(M_l), \quad l = 1,2,\dots,m,$$

which is solved by the method of LS, that is, by assigning in turn the different values from B_m to the variable b and solving the resulting system of linear equations with unknowns in A. The computational complexity of the attack is $\mathrm{O}(\mathrm{C}_n^{n/2})$.

4. $J = \{\sigma_2\} \simeq \{d\}$.

$S_l = g^{-1}(M_l \oplus d)$, $l = 1,2,\dots,m$, and

$$f^{-1}(x) = g^{-1}(x \oplus d)$$

where

$$d = M_l \oplus g(S_l), \quad l = 1,2,\dots,m.$$

Computational complexity of the attack is a polynomial in n.

5. $J = \{\sigma_1,\sigma_2\} \simeq \{b,d\}$.

$S_l = g^{-1}(M_l \oplus d) \oplus b$, $l = 1,2,\dots,m$, and

$$f^{-1}(x) = g^{-1}(x \oplus d) \oplus b$$

where (b,d) is a solution of the following system of equations

$$g(S_l \oplus b) = M_l \oplus d, \quad l = 1,2,\dots,m,$$

which is solved, using the method of LS, that is, by assigning in turn the different values to the variable b and solving the resulting system of linear equations with unknowns in d. The computational complexity of the attack is $\mathrm{O}(2^n)$.

6. $J = \{\pi_1,\sigma_2\} \simeq \{A,d\}$.

$S_l = A^{-1}g^{-1}(M_l \oplus d)$, $A(S_l \oplus d) = g^{-1}(M_l)$, $\mathrm{w}(S_l \oplus d) = \mathrm{w}(g^{-1}(M_l))$, $l = 1,2,\dots,m$, $D_1 = \{\underline{d} : \mathrm{w}(S_1 \oplus \underline{d}) = \mathrm{w}(g^{-1}(M_1))\}$, and $D_l = \{\underline{d} : \underline{d} \in D_{l-1}, \mathrm{w}(S_l \oplus \underline{d}) = \mathrm{w}(g^{-1}(M_l))\}$, $2 \leqslant l \leqslant m$. So,

$$f^{-1}(x) = A^{-1}g^{-1}(x \oplus d)$$

where (A,d) is a solution of the system of equations

$$AS_l = g^{-1}(M_l \oplus d), \quad l = 1,2,\dots,m,$$

which is solved, using the method of LS, that is, by assigning in turn the different values from D_m to the variable d and solving the resulting system of linear equations with unknowns in A. The computational complexity of the attack is $\mathrm{O}(\mathrm{C}_n^{n/2})$.

7. $J = \{\pi_1,\sigma_1,\sigma_2\} \simeq \{A,b,d\}$.

$S_l = A^{-1}g^{-1}(M_l \oplus d) \oplus b$, $A(S_l \oplus b) = g^{-1}(M_l \oplus d)$, $\mathrm{w}(S_l \oplus b) = \mathrm{w}(g^{-1}(M_l \oplus d))$, $l = 1,2,\dots,m$, $B_1(\underline{d}) = \{\underline{b} : \mathrm{w}(S_1 \oplus \underline{b}) = \mathrm{w}(g^{-1}(M_1 \oplus \underline{d}))\}$, and $B_l(\underline{d}) = \{\underline{b} : \underline{b} \in B_{l-1}(\underline{d}),$ $\mathrm{w}(S_l \oplus \underline{b}) = \mathrm{w}(g^{-1}(M_l \oplus \underline{d}))\}$, $\underline{d} \in \mathbb{F}_2^n$, $2 \leqslant l \leqslant m$. So,

$$f^{-1}(x) = A^{-1}g^{-1}(x \oplus d) \oplus b,$$

where (A,b,d) is a solution of the system of equations

$$A(S_l \oplus b) = g^{-1}(M_l \oplus d), \quad l = 1,2,\dots,m,$$

which is solved, using the method of LS, that is, by assigning in turn the different values $(\underline{b},\underline{d})$ with $\underline{d}$ from $\mathbb{F}_2^n$ and $\underline{b}$ from $B_m(\underline{d})$ to the pair of variables (b,d) and solving the resulting system of linear equations with unknowns in A. The computational complexity of the attack is $\mathrm{O}(2^n\mathrm{C}_n^{n/2})$.

8. $J=\{\pi_2\}\simeq\{D\}$.

$S_l=g^{-1}(D^{-1}M_l)$, $l=1,2,\ldots,m$, and

$$f^{-1}(x)=g^{-1}(D^{-1}x)$$

where D^{-1} is a solution of the system of linear equations

$$D^{-1}M_l=g(S_l),\quad l=1,2,\ldots,m.$$

Computational complexity of this attack is a polynomial in n.

9. $J=\{\pi_2,\sigma_1\}\simeq\{D,b\}$.

$S_l=g^{-1}(D^{-1}M_l)\oplus b$, $g(S_l\oplus b)=D^{-1}M_l$, $\mathrm{w}(M_l)=\mathrm{w}(g(S_l\oplus b))$, $l=1,2,\ldots,m$, $B_1=\{\underline{b}:\mathrm{w}(M_1)=\mathrm{w}(g(S_1\oplus\underline{b}))\}$, and $B_l=\{\underline{b}:\underline{b}\in B_{l-1},\mathrm{w}(M_l)=\mathrm{w}(g(S_l\oplus\underline{b}))\}$, $2\leqslant l\leqslant m$. So,

$$f^{-1}(x)=g^{-1}(D^{-1}x)\oplus b$$

where (D^{-1},b) is a solution of the system of equations

$$D^{-1}M_l=g(S_l\oplus b),\quad l=1,2,\ldots,m,$$

which is solved, using the method of LS, that is, by assigning in turn the different values from B_m to the variable b and solving the resulting system of linear equations with unknowns in D^{-1}. The computational complexity of the attack is $\mathrm{O}(\mathrm{C}_n^{n/2})$.

10. $J=\{\pi_1,\pi_2\}\simeq\{A,D\}$.

$S_l=A^{-1}g^{-1}(D^{-1}M_l)$, $l=1,2,\ldots,m$, and

$$f^{-1}(x)=A^{-1}g^{-1}(D^{-1}x)$$

where (A,D^{-1}) is a solution of the system of equations

$$D^{-1}M_l=g(AS_l),\quad l=1,2,\ldots,m,$$

which is solved, using the method of LS, that is, by assigning in turn the different values to the variable A and solving the resulting system of linear equations with unknowns in D^{-1}. So, the computational complexity of the attack is $\mathrm{O}(n!)$.

11. $J=\{\pi_1,\pi_2,\sigma_1\}\simeq\{A,D,b\}$.

$S_l=A^{-1}g^{-1}(D^{-1}M_l)\oplus b$, $l=1,2,\ldots,m$, and

$$f^{-1}(x)=A^{-1}g^{-1}(D^{-1}x)\oplus b$$

where (A,D^{-1},b) is a solution of the system of equations

$$A^{-1}g^{-1}(D^{-1}M_l)\oplus b=S_l,\quad l=1,2,\ldots,m,$$

which is solved, using the method of LS, that is, by assigning in turn the different values to the variables D^{-1} and solving the resulting system of linear equations with unknowns in A,b. The computational complexity of the attack is $\mathrm{O}(n!)$.

12. $J = \{\pi_2, \sigma_2\} \simeq \{D, d\}$.
$S_l = g^{-1}(D^{-1}M_l \oplus d)$, $l = 1, 2, \ldots, m$, and

$$f^{-1}(x) = g^{-1}(D^{-1}x \oplus d)$$

where (D^{-1}, d) is a solution of the system of equations

$$D^{-1}M_l \oplus d = g(S_l), \quad l = 1, 2, \ldots, m,$$

which is solved with a polynomial complexity.

13. $J = \{\pi_2, \sigma_1, \sigma_2\} \simeq \{D, b, d\}$.
$S_l = g^{-1}(D^{-1}M_l \oplus d) \oplus b$, $l = 1, 2, \ldots, m$, and

$$f^{-1}(x) = g^{-1}(D^{-1}x \oplus d) \oplus b$$

where (D^{-1}, d, b) is a solution of the system of equations

$$D^{-1}M_l \oplus d = g(S_l \oplus b), \quad l = 1, 2, \ldots, m,$$

which is solved, using the method of LS, by assigning in turn the different values to the variable b and solving the resulting system of linear equations with unknowns in D^{-1} and d. So, the computational complexity of the attack is $\mathrm{O}(2^n)$.

14. $J = \{\pi_1, \pi_2, \sigma_2\} \simeq \{A, D, d\}$. $S_l = A^{-1}g^{-1}(D^{-1}M_l \oplus d)$, $l = 1, 2, \ldots, m$, and

$$f^{-1}(x) = A^{-1}g^{-1}(D^{-1}x \oplus d)$$

where (D^{-1}, d, A) is a solution of the system of equations

$$D^{-1}M_l \oplus d = g(AS_l), \quad l = 1, 2, \ldots, m,$$

which is solved, using the method of LS, by assigning in turn the different values to the variable A and solving the resulting system of linear equations with unknowns in D^{-1} and d. So, the computational complexity of the attack is $\mathrm{O}(n!)$.

Conclusion

What we have discussed above in the paper should be considered as a step in the process of developing the theory of public key cryptosystems based on the bijective systems of Boolean functions. There are many problems we need solve on this way. Some of them are the following: 1) generating pseudorandom invertible systems of Boolean functions depending on a covered parameter such that the system is computed and inverted with a polynomial time complexity iff the value of the parameter is known; 2) necessary and sufficient conditions for uniqueness of a private key, with which the given blocks of a ciphertext are decrypted to the given blocks of a plaintext; 3) lower and upper bounds for the number of blocks with this property of the private key.

REFERENCES

1. *Tao R.* Finite Automata and Application to Cryptography. Berlin; Heidelberg, Springer, 2009. 411 p.
2. *Agibalov G. P.* Substitution block ciphers with functional keys. Prikladnaya Diskretnaya Matematika, 2017, no. 38, pp. 57–65.
3. *Agibalov G. P.* Metody resheniya sistem polinomial'nykh uravneniy nad konechnym polem [Methods for solving systems of polynomial equations over a finite field]. Vestnik TSU. Prilozhenie, 2006, no. 17, pp. 4–9. (in Russian)
4. *Agibalov G. P.* Logicheskie uravneniya v kriptoanalize generatorov klyuchevogo potoka [Logical equations in cryptanalysis of key stream generators]. Vestnik TSU. Prilozhenie, 2003, no. 6, pp. 31–41. (in Russian)