О криптографической стойкости датчика MQ_DRBG

В.О. Дрелихов, Г.Б. Маршалко, А.В. Покровский

Аннотация

В данной работе анализируется отображение $\mathbb{F}_2^n \to \mathbb{F}_2^{n+r}$, реализуемое датчиком MQ_DRBG, который был предложен Французским агенством по стандартизации (AFNOR) к международной стандартизации в качестве детерминированного датчика псевдослучайных чисел. Датчик реализуется с помощью однонаправленной функции $S : \mathbb{F}_2^n \to \mathbb{F}_2^{n+r}$ специального вида, координатные функции которой являются квадратичными функциями многих переменных. Мы предлагаем два метода построения функций S, удовлетворяющих спецификациям, указанным в проекте стандарта, для которых трудоемкость нахождения аргумента по его образу ниже указанной в проекте.

1 Введение

В 2010 году в рамках пересмотра международного стандарта выработки случайных чисел ISO/IEC 18031 [7] Французским агенством по стандартизации (AFNOR) было предложено дополнить стандарт детерминированным датчиком случайных чисел, задача восстановления начального заполнения которого эквивалентна задаче решения квадратичной системы уравнений [5]. Основными аргументами, высказанными французскими специалистами в качестве обоснования необходимости включения датчика в стандарт и приведенными в работе [4], являются:

- трудоемкость восстановления начального состояния датчика эквивалентна трудоемкости задачи решения системы квадратичных уравнений, о которой известно, что она NP-полна;
- датчик эффективно реализуется с помощью базовых операций AND и XOR;
- датчик MQ_DRBG медленнее стандартизованных датчиков на основе блочных шифров и функций хеширования, но для него может быть получена асимптотическая оценка стойкости с помощью аппарата теории доказуемой стойкости (provable security) при условии случайного выбора системы и начального заполнения;

- датчик MQ_DRBG гораздо быстрее стандартизированных решений на базе математического аппарата теории чисел и алгебраической геометрии.

При доказательстве криптографических свойств датчика автор [3] использовал аппарат теории доказуемой стойкости (provable security), который позволил показать, что датчик удовлетворяет требованиям предъявляемым стандартом ISO/IEC 18031 [7] и широко применяемой методикой AIS20 [8] при условии случайного равновероятного выбора системы функций, задающий датчик, из всего множества систем, удовлетворяющих спецификациям [5], и случайного равновероятного выбора начального заполнения.

Одной из задач, стоящих перед исследователями, является защита от злонамеренного разработчика криптографических средств, реализующего заведомо «слабые» криптографические решения. В случае датчика, задаваемого системой квадратичных функций, такая проблематика стоит особенно остро. Действительно, в настоящее время активно исследуются асимметричные шифрсистемы, также основанные на преобразованиях, задаваемых системами полиномиальных уравнений, при этом, для реализации подобных систем ([10],[11],[12],[13]) необходимо подобрать обратимую систему полиномиальных уравнений, которая затем «маскируется» с помощью секретных аффинных (линейных) преобразований вектора аргументов и вектора значений, не позволяющих аналитику определить, может ли система быть обращена.

Таким образом, важным представляется исследование вопроса о том, возможно ли построение «слабых» систем квадратичных уравнений (трудоемкость решения которых существенно ниже, чем минимальная трудоемкость указанная в [5]), которые затем могут быть преобразованы к «псевдослучайному» виду.

Отметим, что в работе [11] указанная задача «маскирования» обратимых систем формулируется следующим образом. Пусть $F(x) = (f_1(x), \ldots, f_r(x))$ и $G(x) = (g_1(x), \ldots, g_r(x))$ – две системы квадратичных функций. Необходимо определить существуют ли два невырожденных аффинных преобразования S и U удовлетворяющие условию:

$$G(x) = U(F(S(x))),$$

что соответствует аффинному преобразованию вектора аргументов и вектора значений систем функций. Если U – единичная матрица, то сформулированную выше задачу называют задачей с одним секретом (IP1S), в противном случае с двумя (IP2S). К настоящему моменту наиболее существенные продвижения получены в решении задачи IP1S. В частности, в работе [15], в предположении, что известны F и G, предлагается алгоритм восстановления преобразования S, имеющий трудоемкостью $O(n^6)$, где n – число аргументов. Такие предположения выполняются для некоторых асимметричных шифрсистем. Вместе с тем, напрямую эти результаты в рассматриваемом нами случае неприменимы, поскольку исходя из общей постановки аналитику неизвестна исходная «слабая» система.

В настоящей работе мы исследуем возможность построения систем квадратичных уравнений, задающих датчик MQ_DRBG, удовлетворяющих спецификациям [5], но имеющих трудоемкость решения меньше гарантированной авторами. Работа организована следующим образом. В первом разделе вводятся обозначения, основные определения и необходимые теоретические сведения. Во втором разделе описаны подходы к построению систем уравнений, которые могут быть решены с трудоемкостью меньшей, чем указана в [5] с помощью методов опробования части переменных и встречи посередине.

2 Основные определения и обозначения

Обозначим через $\mathcal{F}_{\mathrm{n}}$ – множество булевых функций от n переменных и AGL – группу аффинных преобразований на $V_n = \{0,1\}^n$. Через $\mathrm{wt}(f)$ будем обозначать вес функции $f \in \mathcal{F}_{\mathrm{n}}$. Скалярное произведение двоичных векторов $\alpha, x \in V_n$ будем записывать в виде (α, x).

Определение 1. *Функции $f_1, f_2 \in \mathcal{F}_n$ называются аффинно эквивалентными, если существует пара $(A, \gamma) \in \mathrm{AGL}$ такая, что $f_1(x \cdot A \oplus \gamma) = f_2(x)$ для любого $x \in \mathbb{F}_2^n$. Обозначать аффинную эквивалентность будем как $f_1 \underset{\mathrm{AGL}}{\sim} f_2$.*

Справедлива следующая теорема (см., например, [2],[1]).

Теорема 1. *Пусть f – квадратичная булева функция из $\mathcal{F}_{\mathrm{n}}$, тогда f аффинно эквивалентна одному из перечисленных ниже видов:*

$$q_1^k(x_1, \ldots, x_n) = x_1x_2 \oplus \ldots \oplus x_{2k-1}x_{2k} \oplus 1,$$
$$q_2^k(x_1, \ldots, x_n) = x_1x_2 \oplus \ldots \oplus x_{2k-1}x_{2k},$$
$$q_3^k(x_1, \ldots, x_n) = x_1x_2 \oplus \ldots \oplus x_{2k-1}x_{2k} \oplus x_{2k+1}.$$

Число $2k$, $0 \leqslant k \leqslant [n/2]$ называется рангом квадратичной формы. Канонические формы q_i^k $i = \overline{1,3}$ имеют вес $2^{n-1} \pm 2^{n-k-1}$, 2^{n-1} соответственно и лежат на разных орбитах (см., например, [2]), индуцируемых группой AGL. Множество всех булевых функций степени не выше второй замкнуто относительно операции сложения и образует код называемый кодом Рида – Маллера второго порядка, который мы будем обозначать $\mathrm{RM}(2, n)$. Если обозначить

$$A_i = |\{f \in \mathrm{RM}(2,n) \mid \mathrm{wt}(f) = i\}|,$$

то справедлива теорема.

Теорема 2 ([1])**.** *Весовой спектр кода $\mathrm{RM}(2, n)$ имеет вид $A_0 = A_{2^n} = 1$,*

$$A_{2^{n-1}-2^{n-k-1}} = A_{2^{n-1}+2^{n-k-1}} = 2^{k(k+1)} \frac{(2^n-1)(2^{n-1}-1)\cdot\ldots\cdot(2^{n-2k+1}-1)}{(2^{2k}-1)(2^{2k-2}-1)\cdot\ldots\cdot(2^2-1)}$$

для всех $1 \leqslant k \leqslant [n/2]$. *Остальные* A_i *равны нулю, за исключением* A_{2^n-1}, *который может быть найден с помощью соотношения* $\sum_{i=0}^{2^n} A_i = 2^{\frac{n^2}{2}+\frac{n}{2}+1}$.

3 Описание датчика MQ_DRBG

В рамках данного раздела рассмотрим систему квадратичных булевых функций, порождаемую датчиком MQ_DRBG. Система S состоит из $n + r$ функций от n переменных, каждая из которых имеет вторую степень нелинейности. Тогда она реализует отображение $V_n \to V_{n+r}$ и i-ый такт работы датчика может быть описан следующим образом:

- задается текущее значение инициализирующего вектора $x_i \in V_n$ (неизвестное аналитику);
- на вектор x_i действует преобразование S, в результате чего получаем $S(x_i) = (z||y)$, где $y \in V_n, \quad z \in V_r$;
- $x_{i+1} = y$;
- z поступает в канал связи.

При этом n уравнений определяют функцию перехода внутренних состояний датчика. Эту подсистему будем обозначать $G(x)$ $G : V_n \to V_n$. Оставшиеся r уравнений задают функцию выхода $F(x)$, $F : V_n \to V_r$. Указанные функции представимы в виде

$$F(x) = \begin{pmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_r(x) \end{pmatrix}. \tag{1}$$

$$G(x) = \begin{pmatrix} f_{r+1}(x) \\ f_{r+2}(x) \\ \vdots \\ f_{r+n}(x) \end{pmatrix}. \tag{2}$$

где каждая из координатных функций является квадратичной булевой функцией от n переменных.

Система из $n + r$ уравнений должна удовлетворять следующему условию.

Определение 2. *Будем говорить, что система функций* $f_i(x), i = \overline{1, n+r}$ *удовлетворяет условию AFNOR с параметрами* l_{max} *и* $\rho_{min}(n)$, *если*

- функции $f_i(x), i = \overline{1, n+r}$ *являются квадратичными формами ранга большего или равного* $\rho_{min}(n)$,

- линейная комбинация не более чем l_{max} *функций также имеет ранг больший или равный* $\rho_{min}(n)$.

В [5] авторы предлагают различные варианты параметров системы квадратичных уравнений, имеющих заданную трудоемкость решения. Ниже в таблице приведены соответствующие значения параметров для каждого предложенного варианта системы и значение эквивалентной криптографической стойкости (в каждой ячейке приведены длина n вектора внутреннего состояния, длина r выходного вектора, минимальное значение $\rho_{min}(n)$ ранга, параметр линейной комбинации l_{max}, а также поле, над которым задаются квадратичные формы).

80 2-TDEA	$n = r = 112$ $GF(2)$,$l_{max} = 4$ $\rho_{min}(n) = 106$	$n = r = 128$ $GF(2^4)$,$l_{max} = 5$ $\rho_{min}(n) = 30$	$n = r = 192$ $GF(2^6)$,$l_{max} = 5$ $\rho_{min}(n) = 30$	$n = r = 256$ $GF(2^8)$,$l_{max} = 5$ $\rho_{min}(n) = 30$
112 3-TDEA	$n = 120, r = 112$ $GF(2)$,$l_{max} = 4$ $\rho_{min}(n) = 114$	$n = r = 128$ $GF(2)$,$l_{max} = 5$ $\rho_{min}(n) = 122$	$n = r = 192$ $GF(2^4)$,$l_{max} = 5$ $\rho_{min}(n) = 44$	$n = r = 256$ $GF(2^4)$,$l_{max} = 5$ $\rho_{min}(n) = 60$
128 AES-128	–	$n = r = 128$ $GF(2)$,$l_{max} = 4$ $\rho_{min}(n) = 122$	$n = r = 192$ $GF(2^3)$,$l_{max} = 5$ $\rho_{min}(n) = 60$	$n = r = 256$ $GF(2^4)$,$l_{max} = 5$ $\rho_{min}(n) = 60$
192 AES-192	–	–	$n = 200, r = 192$ $GF(2)$,$l_{max} = 4$ $\rho_{min}(n) = 192$	$n = r = 256$ $GF(2^2)$,$l_{max} = 4$ $\rho_{min}(n) = 124$
256 AES-256	–	–	–	$n = 272, r = 256$ $GF(2)$,$l_{max} = 4$ $\rho_{min}(n) = 264$

Таблица 1: Параметры датчика MQ_DRBG

Далее в этом разделе мы рассмотрим способы построения систем функций, задающих выходную функцию $F(x)$ генератора, которые удовлетворяют указанным в таблице 3 спецификациям, но имеют существенно меньшую оценку трудоемкости решения.

3.1 Метод опробования части переменных

Зафиксируем квадратичную форму

$$q = x_1(\alpha_1, y) \oplus \ldots \oplus x_k(\alpha_k, y),\ y, \alpha_i \in V_k,\ i = \overline{1,k},$$

реализующую булеву функцию от $2k$ переменных и удовлетворяющую условию: набор функций $x_1, \ldots, x_k, (\alpha_1, y), \ldots, (\alpha_k, y)$ линейно независим над $\mathbb{F}_2$.

Невырожденной линейной заменой переменных $x_i = z_{2i-1},\ (\alpha_i, y) = z_{2i}\ \ i = \overline{1,k}$ квадратичную форму q можно привести к виду:

$$q'(z_1, \ldots, z_{2k}) = z_1 z_2 \oplus \ldots \oplus z_{2k-1} z_{2k}$$

ранга $2k$.

Определим отображение $\tau_q \ : \ V_k \to \mathbb{F}_{2^k}$

$$\tau_q(y) = (\alpha_1, y)\,\xi \oplus \ldots \oplus (\alpha_k, y)\,\xi^{2^{k-1}},$$

действующее из V_k в $\mathbb{F}_{2^k}$, где $\left\langle \xi, \xi^2, \ldots, \xi^{2^{k-1}} \right\rangle$ нормальный базис поля $\mathbb{F}_{2^k}$. Определим также отображение $\pi_i \ : \ \mathbb{F}_{2^k} \to \mathbb{F}_{2^k}$ $i = \overline{0, k-1}$ следующим образом:

$$\pi_i(v) = \xi^{2^i} v, \quad v \in \mathbb{F}_{2^k}.$$

Отображение τ_q является линейным и сюръективным. Линейность очевидна, а сюръективность следует из того, что

$$\operatorname{rang} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{pmatrix} = k.$$

В свою очередь, π_i – невырожденное линейное преобразование. Пусть $v = \tau_q(y) \in \mathbb{F}_{2^k}$, тогда

$$\pi_i(v) = (\alpha_1, y)\,\xi^{2^i+1} \oplus \ldots \oplus (\alpha_k, y)\,\xi^{2^i+2^{k-1}} = \left(\alpha_1^{(i)}, y\right)\xi \oplus \ldots \oplus \left(\alpha_k^{(i)}, y\right)\xi^{2^{k-1}}. \qquad (3)$$

В матричной форме имеет место равенство:

$$\begin{pmatrix} \left(\alpha_1^{(i)}, y\right) \\ \vdots \\ \left(\alpha_k^{(i)}, y\right) \end{pmatrix} = C \cdot \begin{pmatrix} (\alpha_1, y) \\ \vdots \\ (\alpha_k, y) \end{pmatrix} = C \cdot \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_k \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_k \end{pmatrix} = \begin{pmatrix} \alpha_1^{(i)} \\ \vdots \\ \alpha_k^{(i)} \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_k \end{pmatrix}, \quad (4)$$

где C – невырожденная матрица размера $k \times k$ над $\mathbb{F}_2$, $\alpha_j,\ \alpha_j^{(i)} \in V_k,\ y_j \in \{0,1\},\ j = \overline{1,k}$.

Вектора $\alpha_1^{(i)}, \ldots, \alpha_k^{(i)}$ – линейно независимы над $\mathbb{F}_2, i = \overline{1,k}$.

Композиция отображений $\pi_i(\tau_q(y)), i = \overline{0,k-1}$ определяет квадратичную форму по правилу:

$$q_i' = x_1(\alpha_1^{(i)}, y) \oplus \ldots \oplus x_k(\alpha_k^{(i)}, y).$$

В силу того, что система функций $x_1, \ldots, x_k, (\alpha_1^{(i)}, y), \ldots, (\alpha_k^{(i)}, y)$ линейно независима над $\mathbb{F}_2$, ранг q_i' равен $2k$. Рассмотрим линейную комбинацию

$$\pi_{i_1}(\tau_q(y)) \oplus \ldots \oplus \pi_{i_d}(\tau_q(y)) = \xi^{2^{i_1}}\tau_q(y) \oplus \ldots \oplus \xi^{2^{i_d}}\tau_q(y) = (\xi^{2^{i_1}} \oplus \ldots \oplus \xi^{2^{i_d}})\tau_q(y).$$

Это преобразование является линейным и невырожденным, в силу того, что $\left\langle \xi, \ldots, \xi^{2^{k-1}} \right\rangle$ – нормальный базис поля $\mathbb{F}_{2^k}$.

Коэффициенты элемента $\oplus_{j=1}^{d} \pi_{i_j}(\tau_q(y))$ в разложении по указанному базису определяют квадратичную форму

$$q'_{i_1 \ldots i_d} = q'_{i_1} \oplus \ldots \oplus q'_{i_d}.$$

При этом ранг $q'_{i_1 \ldots i_d}$ будет равен $2k$ в силу невырожденности порождающего ее преобразования (доказательство этого факта производится аналогично рассмотренному выше случаю $d = 1$). Таким образом, любая ненулевая комбинация $c_1 q'_1 \oplus \ldots \oplus c_k q'_k$ дает квадратичную форму ранга $2k$, $c_j \in \{0, 1\}$, $j = \overline{1, k}$.

Квадратичной форме вида $q_{i_1} \oplus \ldots \oplus q_{i_d}$ поставим в соответствие двоичный вектор из V_k вида

$$(0 \ldots 0 \underbrace{1}_{i_1} 0 \ldots 0 \underbrace{1}_{i_2} 0 \ldots 0 \underbrace{1}_{i_d} 0 \ldots 0)$$

Рассмотрим линейный код C длины k, размерности m, такой что $2^m - 1 \geq n + r$ и любая линейная комбинация из не более чем 4-х кодовых слов не равна нулю. Если существует код $C^{\perp}$ с параметрами $[k, k - m, d]$, где $d > 4$, то можно построить квадратичную систему из $2^m - 1$ уравнения, каждое из которых имеет вид:

$$x_1(\alpha_1^{(i)}, y) \oplus \ldots \oplus x_k(\alpha_k^{(i)}, y) \oplus (\beta, z) = \gamma,$$

где $\gamma \in \{0, 1\}$, $\beta, z \in V_t$ и координаты векторов $(x_1, \ldots, x_k)$, y, z являются переменными системы.

Метод решения предложенной системы заключается в переборе значений переменных $x_1, \ldots, x_k$, что требует 2^k операций, после чего каждое квадратичное уравнение становится линейным, если не все переменные зафиксированы нулями. В результате для каждого варианта зафиксированных переменных (кроме нулевого) будет получаться линейная система от k оставшихся неизвестных, которую можно решить с трудоемкостью порядка k^3 двоичных операций, и тем самым восстановить инициализирующий вектор. Общая трудоемкость метода составит порядка $2^k k^3$ двоичных операций.

Трудоемкость такого подхода, например, при $k = 53$ (соответствующий линейный код при этом значении параметра существует [2]) составляет $2^{53}(59)^3 \approx 2^{71}$ двоичных операций, что на девять порядков меньше обозначенной в проекте стандарта трудоемкости.

3.2 Метод встречи посередине

Рассмотрим возможность применения метода встречи посередине для восстановления начального состояния генератора. Для применения метода

необходимо, чтобы выходная вектор-функция датчика могла быть представлена в виде суммы по крайней мере двух вектор-функций от независимых переменных.

Выберем s квадратичных форм $h_i(z_1, \ldots, z_{n/2}), i = \overline{1, s}$ от $n/2$ переменных, удовлетворяющих условию AFNOR с параметрами l_{max} и $\rho_{min}(n/2) \geq \rho_{min}(n)/2$ и составим систему функций, задающих генератор, следующим образом.

$$\begin{pmatrix} F(x) \\ G(x) \end{pmatrix} = \begin{pmatrix} h_1(x_1, \ldots, x_{n/2}) \oplus h_1(x_{n/2+1}, \ldots, x_n) \\ h_2(x_1, \ldots, x_{n/2}) \oplus h_2(x_{n/2+1}, \ldots, x_n) \\ \vdots \\ h_s(x_1, \ldots, x_{n/2}) \oplus h_s(x_{n/2+1}, \ldots, x_n) \\ f_{s+1}(x) \\ \vdots \\ f_{n+r}(x) \end{pmatrix}, \tag{5}$$

В этой системе первые s функции образованы суммой выбранных функций от независимых переменных и удовлетворяют условию AFNOR с параметрами l_{max} и $\rho_{min}(n)$. Оставшиеся функции выберем случайно, так чтобы выполнялось условие AFNOR с параметрами l_{max} и $\rho_{min}(n)$.

Оценим сложность реализации метода встречи посередине. На первом этапе опробуем аргументы вектор-функции $F' : V_{n/2} \to V_s$

$$F'(x_1, \ldots, x_{n/2}) = \begin{pmatrix} h_1(x_1, \ldots, x_{n/2}) \\ h_2(x_1, \ldots, x_{n/2}) \\ \vdots \\ h_s(x_1, \ldots, x_{n/2}) \end{pmatrix},$$

и по адресу, равному значению функции $F'(x)$, запишем векторы аргументов, которым соответствует данное значение. Этот этап потребует $2^{n/2}$ операций вычисления функции $F'(x)$, а необходимый объем памяти $(n/2) * 2^s * 2^{n/2-s} = (n/2) * 2^{n/2}$.

Пусть теперь нам известен выходной вектор, полученный с MQ_DRBG, $y = (y_1, \ldots, y_s, y_{s+1}, \ldots, y_r)$. На втором этапе метода для каждого аргумента функции $(x^i_{n/2+1}, \ldots, x^i_n), i = \overline{0, 2^{n/2} - 1}$ вычислим вектор $(v_1, \ldots, v_s) = (y^i_1, \ldots, y^i_s) \oplus F'(x^i_1, \ldots, x^i_{n/2})$. Далее переберем все находящиеся по адресу $(v_1, \ldots, v_s)$ аргументы $(x_1, \ldots, x_{n/2})$ и вычислим значение функции выхода генератора $F(x^i_1, \ldots, x^i_s, x_{n/2+1}, \ldots, x_n)$. В случае, если полученное значение совпадает с выходным вектором, найдено истинное начальное заполнение. Второй этап потребует в среднем $2^{n/2}2^{n/2-s}$ операций, при условии, что отображение, задающее выходной вектор MQ_DRBG близко по своим характеристикам к случайному.

В конечном итоге метод потребует $2^{n/2} + 2^{n/2}2^{n/2-s}$ операций вычисления функции F или, учитывая, что операция умножения вектора на матрицу реализуется за n^2 двоичных операций, $2^{n/2} * s * 2n^2 + 2^{n/2}2^{n/2-s} * r * 2n^2$ двоичных операций.

В рамках исследований проведены программные эксперименты, которые показали, что для некоторых значений параметров датчика MQ_DRBG , указанных в [5], указанные системы функций могут быть построены с помощью случайного выбора.

В частности, для параметров датчика $n = 200$, $r = 192$ была построена система при $s = 95$, которая может быть решена с трудоемкостью 2^{129}, что на 63 двоичных порядка меньше указанной авторами;

Также отметим, что, с использованием предложенного в предыдущем разделе метода нормальных базисов, для некоторых значений параметров датчика можно строить предложенные системы для произвольного s.

4 Выводы

Предложенный французскими специалистами вариант датчика псевдослучайных чисел реализует отображение, задаваемое системой квадратичных функций, к которым предъявляется ряд требований, определяемых спецификацией C.5 проекта стандарта [5]. Трудоемкость восстановления начального заполнения данного датчика основана на трудоемкости решения задающей его системы. Поскольку в [5] процедура выработки такого рода систем не стандартизирована, в рамках проведенных исследований были предложены два подхода к построению систем квадратичных уравнений, задающих датчик и имеющих трудоемкость решения меньшую, чем минимальная сложность, заявленная авторами проекта стандарта:

- Первый подход основан на построении системы уравнений, которая при опробовании части переменных сводится к линейной. Задача построения систем такого вида, сводится к построению линейного кода с заданными параметрами. Для некоторых параметров датчика такие коды существуют.
- Второй подход основан на применении метода встречи посередине относительно части уравнений. Для некоторых параметров датчика такие системы могут быть получены посредством случайного поиска с ограничениями.
- Предложенные методы могут быть использованы совместно.

Построенные примеры позволяют говорить о том, что требования, предъявляемые проектом стандарта [5] к выбору системы уравнений, не гарантируют заявленную в [5] нижнюю оценку трудоемкости ее решения. Если нарушитель имеет возможность навязать систему, удовлетворяющую требованиям спецификации, но не обеспечивающую задаваемую трудоемкость решения, то эта атака может быть эффективной. Для маскирования построенных «слабых» систем может использован подход, основанный на задаче изоморфизма многочленов.

В конечном итоге по результатам проведенных исследований можно утверждать, что наибольшим вопросом при оценке трудоемкости решения системы уравнений задающей датчик MQ_DRBG является то, что при ее обосновании авторы

исходили из оценки трудоемкости применения общих методов анализа нелинейных систем уравнений. По нашему мнению такой подход, наряду с тем, что авторы фактически рассматривают семейство генераторов (семейство систем уравнений удовлетворяющим заданным свойствам), несет определенную опасность в смысле недостижения заявленной оценки трудоемкости. В связи со сказанным, необходимо отметить, что стойкость уже стандартизованных решений, в частности датчиков на основе блочного шифра и функции хеширования, основана главным образом на стойкости реализованного в них криптографического алгоритма, которая в свою очередь основывается не на оценке трудоемкости применения общих методов анализа, а на невозможности применения методов, использующих особенности алгоритмов.

Список литературы

[1] Логачев О.А., Сальников А.А., Ященко В.В.: *Булевы функции в теории кодирования и криптологии* – М.: МЦНМО, 2004.

[2] Мак-Вильямс Ф. Дж., Слоэн Н. Дж. А.: *Теория кодов исправляющих ошибки* —М.: Связь, 1979.

[3] Paillier P.: *A comprehensive security analysis of* MQ_DRBG. *Technical report TR-017.*

[4] MQ_DRBG. *An AFNOR proposal for ISO/IEC 18031* SC27 WG2 Meeting. Berlin Oct 4-8 2010.

[5] ISO/IEC JTC 1/SC 27 №9808. Text for ISO/IEC FDIS 18031 – Information technology – Security techniques – Random Bit Generation.

[6] Berbain C., Gilbert H. Patarin J.: *QUAD: A practical stream cipher with provable security* // EUROCRYPT'06, vol. 4004 LNCS, 109-128.

[7] ISO/IEC 18031:2005 – Information technology – Security techniques – Random Bit Generation.

[8] AIS20 Version 1. Functionality classes and evaluation methodology for deterministic random number generators. BSI. 1999.

[9] Yang B., Chen O., Bernstein D., Chen J.: *Analysis of QUAD* // FSE 2007, vol. 4593 LNCS, pp. 290 – 308, Springer, 2007.

[10] Matsumoto T., Imai H.: *Public quadratic polynomial-tuples for efficient signature-verification and message-encryption*, In. Advances in Cryptology – EUROCRYPT'88. Volume 330 in LNCS. Springer-Verlag (1988) 419-453.

[11] Patarin J.: *Hidden Field Equations (HFE) and Isomorphism of Polynomials (IP): Two new families of asymmetric algorithms.* In EUROCRYPT'96.

[12] Patarin J., Courtois N., Goubin L.: *Flash, a fast multivariate signature algorithm.* In Proceedings CT-RSA 2001, vol. 2020 of LNCS, Springer, 2001.

[13] Patarin J., Courtois N., Goubin L.: *QUARTZ, 128-bit long digital signatures.* In Proceedings CT-RSA 2001, vol. 2020 of LNCS, Springer, 2001.

[14] Perret L.: *A fast cryptanalysis of the isomorphism of polynomials with one secret problem.* In Proceedings EUROCRYPT'98.

[15] Bouillaguet C., Faugere J.-C.: *Practical cryptanalysis of the identification.* In Proceedings PKC'11.

[16] Faugere J.-C., Perret L.: *Polynomial equivalence problem: Algorithmic and theoretical aspects.* In Proceedings EUROCRYPT'06.