Секция 3

МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

УДК 519.7 DOI 10.17223/2226308X/9/17

К КРИПТОАНАЛИЗУ ДВУХКАСКАДНЫХ КОНЕЧНО-АВТОМАТНЫХ КРИПТОГРАФИЧЕСКИХ ГЕНЕРАТОРОВ

Г. П. Агибалов, И. А. Панкратова

Сообщается о конечно-автоматном обобщении регистрового генератора (δ,τ)-шагов и об атаках на него с угрозой раскрытия начальных состояний и функций выходов автоматов и со сложностью, много меньшей сложности атаки грубой силы.

Ключевые слова: *конечный автомат, криптографический генератор, генератор (δ,τ)-шагов, криптоанализ, линеаризационная атака.*

Рассматриваемый здесь криптографический генератор (будем обозначать его G) представляет собой последовательное соединение двух абстрактных конечных автоматов над полем $\mathbb{F}_2$ — управляющего и управляемого. Первый является некоторым автономным автоматом $A_1 = (\mathbb{F}_2^n, \mathbb{F}_2, g_1, f_1)$ с множеством состояний $\mathbb{F}_2^n$, $n > 1$, выходным алфавитом $\mathbb{F}_2$, с функцией переходов $g_1 : \mathbb{F}_2^n \to \mathbb{F}_2^n$ и функцией выходов $f_1 : \mathbb{F}_2^n \to \mathbb{F}_2$, а второй — некоторым неавтономным автоматом $A_2 = (\mathbb{F}_2, \mathbb{F}_2^m, \mathbb{F}_2, g_2, f_2)$ с входным и выходным алфавитами $\mathbb{F}_2$, с множеством состояний $\mathbb{F}_2^m$, $m > 1$, с функцией выходов $f_2 : \mathbb{F}_2 \times \mathbb{F}_2^m \to \mathbb{F}_2$ и функцией переходов $g_2 : \mathbb{F}_2 \times \mathbb{F}_2^m \to \mathbb{F}_2^m$, для которой существуют отображение $g : \mathbb{F}_2^m \to \mathbb{F}_2^m$ и различные целые неотрицательные числа δ и τ, такие, что для всех $u \in \mathbb{F}_2$ и $y \in \mathbb{F}_2^m$ если $u = 0$, то $g_2(u, y) = g^\delta(y)$, и если $u = 1$, то $g_2(u, y) = g^\tau(y)$, т. е. $g_2(u, y) = \neg u g^\delta(y) + u g^\tau(y)$, где, как обычно, $g^0(y) = y$ и $g^r(y) = g(g^{r-1}(y))$. Таким образом, генератор G — это конечный автономный автомат $G = A_1 \cdot A_2 = (\mathbb{F}_2^n \times \mathbb{F}_2^m, \mathbb{F}_2, h, f)$, где $h : \mathbb{F}_2^n \times \mathbb{F}_2^m \to \mathbb{F}_2^n \times \mathbb{F}_2^m$ и $h(x, y) = (g_1(x), g_2(f_1(x), y))$, $f(x, y) = f_2(f_1(x), y)$, $x \in \mathbb{F}_2^n$, $y \in \mathbb{F}_2^m$. Все функции в определении G, как видим, булевы, причём функции g_1 и g_2, g — векторные размерности n и m соответственно.

Генератор G функционирует в дискретном времени $t = 1, 2, \ldots$, в каждый момент t которого его автомат A_1, находясь в состоянии $x(t) = x_1(t)x_2(t)\ldots x_n(t) \in \mathbb{F}_2^n$, выдаёт выходной управляющий символ $u(t) = f_1(x(t))$ и переходит в следующее состояние $x(t + 1) = g_1(x(t))$, а автомат A_2 в этот момент, находясь в состоянии $y(t) = y_1(t)y_2(t)\ldots y_m(t) \in \mathbb{F}_2^m$, принимает от A_1 символ $u(t)$, выдаёт свой выходной символ $v(t) = f_2(u(t), y(t))$ и переходит в следующее состояние $y(t+1) = g_2(u(t), y(t))$. Значением $z(t)$ на выходе генератора G в момент t является значение $v(t)$ на выходе автомата A_2 в этот момент.

Предполагается, что значение любой функции в генераторе G на любом наборе значений её аргументов вычисляется за полиномиальное время от числа последних.

Можно показать, что в частном случае, когда функции g_1 и g являются функциями переходов некоторых регистров сдвига с линейной обратной связью и длиной n и m соответственно и $f_2(u, y_1, y_2, \ldots, y_m) = y_1$, генератор G функционально эквивалентен генератору (δ,τ)-шагов [1].

Функционирование генератора G во времени описывается формально следующей системой векторных булевых уравнений с переменными $x(t), y(t), u(t)$, $t = 1, 2, \ldots$:

$$\begin{cases} u(t) = f_1(x(t)), \\ x(t+1) = g_1(x(t)), \\ x(1) = x_1(1)x_2(1)\ldots x_n(1); \\ z(t) = f_2(u(t), y(t)), \\ y(t+1) = \neg u(t) g^{\delta}(y(t)) + u(t) g^{\tau}(y(t)), \\ y(1) = y_1(1)y_2(1)\ldots y_m(1); \\ t \geqslant 1. \end{cases}$$

Здесь подсистема E_1 из первого, второго и третьего уравнений описывает работу управляющего автомата A_1, подсистема E_2 из четвёртого, пятого и шестого уравнений — работу управляемого автомата A_2.

Ключом генератора G теоретически может быть любое непустое подмножество множества $\{x(1), y(1), f_1, g_1, f_2, g_2, g, \delta, \tau\}$. Требование стойкости криптографического генератора накладывает определённые ограничения на применяемые в нём булевы функции. Кроме того, не любую булеву функцию можно задать практически. В этой связи предполагается, что каждая функция в генераторе принадлежит некоторому классу, и этот класс, в том числе для функции в составе ключа, общеизвестен. При известном ключе и заданных других параметрах генератора уравнения данной системы позволяют однозначно вычислить порождаемую генератором выходную последовательность $z(1)z(2)\ldots$

Задача криптоанализа произвольного генератора G заключается в определении его ключа в известном классе по заданному конечному отрезку $\gamma = z(1)z(2)\ldots z(l)$ последовательности, порождаемой им на этом ключе. В такой постановке задача возникает в криптоанализе поточного шифра, использующего данный генератор, атакой на шифр с известным или выбираемым открытым текстом. Её решение может быть получено как решение конечной подсистемы S_l указанной системы уравнений с $t = 1, 2, \ldots, l$. В случае неединственности решения системы S_l обычно рекомендуется задаться более длинным отрезком γ.

Решение системы S_l может быть найдено атакой грубой силы, или исчерпывающим поиском, т. е. перебором возможных ключей с вычислением при каждом выбранном ключе k начального отрезка γ' длины l порождаемой последовательности и сравнением его с отрезком γ. В случае $\gamma' = \gamma$ ключ k принимается за ответ задачи. Сложность этой атаки определяется размером ключевого пространства. Так, если ключом генератора служит его начальное состояние, то сложность атаки равна 2^{n+m}. Если же ключом является какая-либо из функций генератора, то сложность атаки равна мощности класса этой функции.

В данной работе показано, что в генераторе G с линейным автоматом A_2 ключ $y(1)$ вскрывается с полиномиальной сложностью решением системы линейных уравнений, а ключ $(x(1), y(1))$ — линеаризационной атакой сложности не более 2^n. Предложен метод, позволяющий в произвольном генераторе G с известными g, δ, τ и f_2 вычислить по γ отрезок управляющей последовательности $\alpha = u(1)u(2)\ldots u(l-1)$ на выходе A_1 и тем самым открыть две возможности для криптоанализа такого G: 1) вычислить его ключ $(x(1), y(1))$ атакой «встреча посредине» со сложностью 2^m; 2) свести задачу криптоанализа G к криптоанализу автомата A_1 — найти его ключ по α. Сложность

метода полиномиальная, если $y(1)$ не входит в ключ, и не превосходит 2^m в противном случае. Ключ $x(1)$ автомата A_1 находится решением его системы уравнений E_1, а вскрытие ключа f_1 в A_1, в свою очередь, сводится к доопределению частичной булевой функции со значениями $u(t)$ на состояниях $x(t)$ для $t = 1, 2, \ldots, l-1$ до функции в классе функции f_1. Аналогично, к доопределению частичной булевой функции со значениями $z(t)$ на парах $(u(t), y(t))$ для $t = 1, 2, \ldots, l$ до функции в классе функции f_2 сводится вскрытие ключа f_2 произвольного генератора G. Осуществление подобного доопределения демонстрируется на классе, состоящем из всех булевых функций от большого числа переменных с малым количеством существенных аргументов из них.

ЛИТЕРАТУРА

1. *Фомичёв В. М.* Дискретная математика и криптология. М.: ДИАЛОГ-МИФИ, 2003. 400 с.

УДК 519.7 DOI 10.17223/2226308X/9/18

О ГРУППЕ, ПОРОЖДЁННОЙ РАУНДОВЫМИ ФУНКЦИЯМИ АЛГОРИТМА БЛОЧНОГО ШИФРОВАНИЯ «КУЗНЕЧИК»

В. В. Власова, М. А. Пудовкина

Одно из направлений исследований итерационных алгоритмов блочного шифрования заключается в описании свойств группы, порожденной множеством всех частичных раундовых функций. Алгоритм «Кузнечик» является новым российским стандартом блочного шифрования. Доказывается, что группа, порождённая множеством всех частичных раундовых функций алгоритма блочного шифрования «Кузнечик», является знакопеременной.

Ключевые слова: *«Кузнечик», ГОСТ Р 34.12-2015, знакопеременная группа.*

Частичные раундовые функции многих алгоритмов блочного шифрования, появившихся в последние годы, представимы в виде композиции преобразований, реализующих слой наложения ключа (X-слой), слой s-боксов (S-слой) и линейный слой (L-слой). Такие алгоритмы называются XSL-алгоритмами блочного шифрования. XSL-алгоритмом является американский стандарт блочного шифрования AES и российский стандарт блочного шифрования «Кузнечик», вступивший в силу в январе 2016 г. Групповым свойствам XSL-алгоритмов посвящены работы [1 – 4]. В [1] для алгоритма AES доказывается, что группа G, порождённая всеми частичными раундовыми функциями, совпадает со знакопеременной. В [3] получены условия, достаточные для примитивности группы G XSL-алгоритма, и доказано, что AES удовлетворяет данным условиям. В [4, 2] получены достаточные условия того, что группа G XSL-алгоритма равна знакопеременной. В [5] приведено исследование приложения группового подхода к построению и анализу криптографических систем.

В данной работе доказывается, что группа, порождённая множеством всех частичных раундовых функций алгоритма «Кузнечик», совпадает со знакопеременной. Для этого используется теорема 1 из [2].

Рассмотрим итерационный алгоритм блочного шифрования. Частичная r-раундовая функция шифрования f_k представима в виде композиции r частичных раундовых функций $g_{k_1}, \ldots, g_{k_r}$, где k_i — раундовый ключ из множества ключей шифрования i-го раунда $K^{(i)}$, $i = 1, \ldots, r$. Во многих алгоритмах множества раундовых ключей совпадают, потому далее в тексте $K^{(1)} = \ldots = K^{(r)} = K$. В работе рассматривается группа $G = \langle g_k \mid k \in K \rangle$.