ПЕРСПЕКТИВНЫЕ МЕТОДЫ ГЕНЕРАЦИИ КЛЮЧЕЙ И КРИПТОГРАФИЧЕСКИХ ПРЕОБРАЗОВАНИЙ

УДК 004.056.55

УСКОРЕННЫЙ МЕТОД ВЫЧИСЛЕНИЯ АЛГЕБРАИЧЕСКОЙ ИМУННОСТИ НЕЛИНЕЙНЫХ УЗЛОВ ЗАМЕНЫ СИММЕТРИЧНЫХ ШИФРОВ

К. Е. ЛИСИЦКИЙ, А. А. КУЗНЕЦОВ, Ю. И. ГОРБЕНКО, В. В. ОНОПРИЕНКО, И. В. СТЕЛЬНИК

Рассматривается ускоренный метод вычисления алгебраической иммунности нелинейных узлов замены симметричных шифров по Жан-Шарлю Фожеру (Jean-Charles Faugère). Он основан на поиске аннигилирующей функции к полиному Жегалкина, построенному от исходного нелинейного узла замены. Приводится анализ быстродействия известного алгоритма подсчёта алгебраической иммунности. Обсуждаются детали реализации алгоритма и приводится описание ускоренного алгоритма вычисления алгебраической иммунности, оптимизированного по времени вычисления и по ресурсам задействования ОЗУ

Ключевые слова: симметричный шифр, алгебраический иммунитет, нелинейный узел замены, булева функция, производительность алгоритма.

ВВЕДЕНИЕ

Важное место среди криптографических алгоритмов защиты информации занимают симметричные (блочные и поточные) шифры. Благодаря своей простоте, быстродействию и надежности они широко используются во многих криптографических приложениях [1–4].

Исследование и анализ симметричных криптосистем, а также их отдельных составных узлов, является актуальной и важной научно-технической задачей. В этой статье речь пойдёт об одном из важнейших примитивов любого симметричного шифра – нелинейном узле замены или, так называемом, S-блоке. Процедура прохождения S-блокового преобразования в шифре обеспечивает дополнительное рассеивание и перемешивание байтов текста, что существенным образом усложняет реализацию различных криптоаналитических атак [3]. А точнее, речь пойдёт об алгебраической иммунности, как об одном из криптографических показателей S-блоков, позволяющих аналитически оценить стойкость симметричных шифров к алгебраическим атакам [5–10]. Основная идея алгебраических атак строится на поиске возможности описания шифрующего преобразования с помощью системы уравнений, связывающих между собой биты открытого текста, ключа и шифртекста [5–7].

Как показывает анализ, расчёт алгебраической иммунности является нетривиальной задачей [11-13]. В частности, для расчета алгебраической иммунности в терминах, введенных Жан-Шарлем Фожером (Jean-Charles Faugère) [11], необходимо оценить минимальную степень полинома из минимального редуцированного базиса Грёбнера идеала, заданного системой уравнений, описывающих нелинейный узел замен. В [13] приведены примеры расчета алгебраической иммунности нелинейных узлов некоторых блочных симметричных шифров. Построение минимального редуцированного базиса Грёбнера при этом реализовано с помощью системы компьютерной алгебры «Magma» [14].

Другой подход к расчету алгебраической иммунности S-блока (векторного отображения) может быть реализован через нахождение минимальной степени ненулевых аннигиляторов булевой функции, описывающей этот S-блок [13]. Действительно, в работе [15, с. 337] показано, что произвольный S-блок с n входами и m выходами может быть однозначно представлен булевой функцией от $n+m$ переменных и алгебраическая иммунность векторного отображения (S-блока) совпадает с минимальной степенью ненулевых аннигиляторов этой функции. Расчет алгебраической иммунности через поиск ненулевых аннигиляторов связан с решением большеразмерной системы уравнений, требующей значительных вычислительных ресурсов и объёма памяти, выходящих за практически приемлемые границы.

В данной работе ставится задача изучить метод расчёта алгебраической иммунности, основанный на решении систем линейных уравнений для нахождения ненулевых аннигиляторов к функции в виде полинома Жегалкина, построенной по исходному S-блоку [13, 15]. В работе предлагается алгоритм расчёта алгебраической иммунности, дается оценка сложности в сравнении с известными результатами.

1. ОБЗОР ЛИТЕРАТУРЫ

Интерес, вызванный в своё время к алгебраическим методам криптоанализа блочных и поточных шифров в 2003 г. работами N. Courtois и W. Meier [1]

не утихает и сегодня. Результатом повышения внимания к этим методам стало появление нового криптографического показателя эффективности S-блоков – алгебраической иммунности. В работе [6] напоминается, что понятие алгебраической иммунности для булевых функций было введено в 2004 г. W. Meier, E. Pasalic и C. Carlet в [7]. Алгебраической иммунностью $AI(f)$ булевой функции $f: Z_2^n \to Z_2$ называется минимальное число d такое, что существует булева функция g степени d, не тождественно равная нулю, для которой $fg = 0$ или $(f \oplus 1)g = 0$ [3, 16]. Булева функция $g \in V_n$ называется *аннигилятором* функции $f \in V_n$, множество различных аннигиляторов образует линейное пространство $Ann(f) = \{g \in V_n \mid f \cdot g = 0\}$ [16].

Далее процитируем ещё одну выдержку из работы [3]. Понятие алгебраической иммунности различными способами было обобщено на векторный случай. Так, в работе [9] F. Armknecht и M. Krause, а также G. Ars и J.-C. Faug`ere в [10] рассмотрели алгебраическую иммунность S-блоков и ввели понятия базовой $AI(F)$ и графической $AI\mathrm{gr}(F)$ алгебраической иммунности векторных булевых функций. При этом базовая алгебраическая иммунность больше 1 только при малых значениях m, поэтому данный параметр анализируется у S-блоков, которые используются в поточных шифрах. Графическая алгебраическая иммунность используется для изучения сопротивляемости алгебраическим атакам блочных шифров. Следующее обобщение, которое воспринимается многими исследователями одним из наиболее естественных с криптографической точки зрения, это компонентная алгебраическая иммунность. Компонентной алгебраической иммунностью $AI_{\mathrm{comp}}(F)$ векторной булевой функции F: $Z_2^n \to Z_2^m$ называется минимальная алгебраическая иммунность компонентных функций $b\,F$ ($b \in Z_2^m$, $b \neq 0$), т. е. $AI_{\mathrm{comp}}(F) = \min\{AI(b\,F) : b \in Z_2^m,\ b \neq 0\}$, где $b\,F = b_1 f_1 \oplus \ldots \oplus b_m f_m$ [10]. В случае компонентной алгебраической иммунности в [10] также получено, что $AI_{\mathrm{comp}}(F) \le \lfloor n/2 \rfloor$.

Ещё один метод определения алгебраической иммунности нелинейных узлов замены по Жан-Шарлю Фожеру, строится с применением базисов Грёбнера [17]. Предположим, что S-блок задается системой алгебраических уравнений над двоичным полем:

$$\begin{cases} f_1(x_1, x_2, \ldots, x_n) = y_1, \\ f_2(x_1, x_2, \ldots, x_n) = y_2, \\ \ldots \\ f_m(x_1, x_2, \ldots, x_n) = y_m, \end{cases} \quad (1)$$

т.е. совокупностью булевых многочленов

$$\begin{aligned} & y_1 - f_1(x_1, x_2, \ldots, x_n), \\ & y_2 - f_2(x_1, x_2, \ldots, x_n), \\ & \ldots, \\ & y_m - f_m(x_1, x_2, \ldots, x_n) \end{aligned} \quad (2)$$

в кольце $K[x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m]$ от переменных $x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m$ с коэффициентами над полем $K = GF(2)$.

С системой уравнений (1), алгебраически задающих структуру S-блока, свяжем идеал $I(S)$ в кольце $K[x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m]$ (обозначается как $I(S) \lhd K[x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m]$), порожденный многочленами (2):

$$I(S) = (y_1 - f_1, y_2 - f_1, \ldots, y_m - f_m) = \left\{ \begin{matrix} (y_1 - f_1) \cdot r_1 + (y_2 - f_1) \cdot r_2 + \ldots + (y_m - f_m) \cdot r_m; \\ r_1, r_2, \ldots, r_m \in K[x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m] \end{matrix} \right\}.$$

Алгебраическая иммунность $AI(S)$ нелинейного узла (S-блока) определяется как минимальная степень многочлена P из идеала $I(S)$ [11]:

$$AI(S) = \min\{\deg(P), P \in I(S) \lhd K[x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m]\}, \quad (3)$$

причем минимальный редуцированный базис Грёбнера идеала $I(S)$ при степенном обратном словарном упорядочении (degrevlex) содержит линейный базис полиномов P из $I(S)$, таких, что $AI(S) = \deg(P)$. Другими словами, для вычисления алгебраической иммунности $AI(S)$ достаточно построить минимальный редуцированный базис Грёбнера идеала $I(S)$, заданного уравнениями (2) и найти многочлен минимальной степени среди элементов этого базиса.

Связь алгебраической иммунности S-блока и булевой функции показана в [15, с. 337]. Рассмотрим булевую функцию

$$f_S(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m) : GF(2)^{n+m} \to GF(2),$$

значения которой определим следующим образом:

$$f_S(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m) = \begin{cases} 1, \forall i, j : f_i(x_1, x_2, \ldots, x_n) = y_j, \\ 0, \forall i, j : f_i(x_1, x_2, \ldots, x_n) \neq y_j. \end{cases} \quad (4)$$

Множество решений уравнения

$$f_S(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m) - 1 = 0$$

совпадает с множеством решений системы (1). Следовательно, имеем различные базисы $(f_S - 1)$ и $(y_1 - f_1, y_2 - f_1, \ldots, y_m - f_m)$ одного идеала эквивалентных систем, т.е.

$$I(f_S - 1) = I(y_1 - f_1, y_2 - f_1, ..., y_m - f_m).$$

Идеал пространства аннигиляторов $Ann(f_S)$ в кольце $K[x_1, x_2, ..., x_n, y_1, y_2, ..., y_m]$ совпадает с идеалом $I(f_S - 1)$, следовательно, алгебраическая иммунность (3) булевого отображения $S: GF(2)^n \to GF(2)^m$ совпадает с минимальной степенью ненулевых полиномов, принадлежащих аннигилятору функции f_s: $AI(S) = \min\{D\deg(g) \mid g \in Ann(f_S)\}$ [13].

Исследованию различных алгоритмов расчета алгебраической иммунности нелинейных узлов замены и посвящена данная работа.

2. АЛГОРИТМЫ РАСЧЕТА АЛГЕБРАИЧЕСКОГО ИММУНИТЕТА

Прежде чем привести конкретные алгоритмы вычисления алгебраической иммунности, приведём некоторые обозначения, которые вводятся в работах [8, 16].

Моном (одночлен) относительно переменных $x_1, ..., x_n$ запишем в виде

$$x^u = \prod_{i=1}^{n} x_i^{u_i} = \begin{cases} x_i, u_i = 1, \\ 1, u_i = 0, \end{cases} \quad (5)$$

где вектора $x, u \in V_2^n, x = (x_1, ..., x_n), u = (u_1, ..., u_n)$.

Степень одночлена x^u определяется весом Хемминга (числом ненулевых координат) $w_h(u)$ вектора $u = (u_1, ..., u_n)$, т.е.

$$Deg(x^u) = w_h(u).$$

С учетом этих обозначений булеву функцию $f(x)$ в алгебраической нормальной форме (в форме полинома Жегалкина) запишем в виде

$$f(x) = \sum_{u \in GF(2)^n} a_u x^u, \; a_u \in GF(2). \quad (6)$$

Функцию (аннигилятор) $g \in A_d^n(f)$ также представим в виде полинома Жегалкина

$$g(x) = \sum_{v \in GF(2)^n : w_h(v) \le d} b_v x^v, \quad (7)$$

где $b_v \in GF(2)$ – неизвестные коэффициенты аннигилятора, $w_h(v)$ – вес Хемминга вектора $v = (v_1, ..., v_n)$.

Линейное пространство аннигиляторов степени $\le d$ обозначим

$$A_d^n(f) = \{g \in V_n \mid f \cdot g = 0, Deg(g) \le d\} \subset Ann(f).$$

Функция $g(x)$ принадлежит пространству $A_d^n(f)$ только в том случае, если для любого $x \in GF(2)^n$ выполняется равенство $f(x) \cdot g(x) = 0$.

Приведём алгоритм вычисления алгебраической иммунности булевой функции [16] и оценим его возможную стандартную реализацию по объёму вычислений.

Алгоритм вычисления алгебраической иммунности булевой функции

Вход: $n \in N$, функция $f(x)$ (заданная списком одночленов x^u с ненулевыми коэффициентами a_u в (6)).

Выход: Значение алгебраической иммунности $AI(f)$.

Шаг 1. Присваиваем $d = 1$.

Шаг 2. Вычисляем пространство аннигиляторов $A_d^n(f)$ и $A_d^n(f + 1)$.

Шаг 3. Если $A_d^n(f) = 0$ и $A_d^n(f + 1) = 0$ присваиваем $d = d + 1$ и переходим к шагу 2.

Шаг 4. Если $A_d^n(f) \ne 0$ и/или $A_d^n(f + 1) \ne 0$ присваиваем $AI(f) = d$ и подаем на выход алгоритма.

Для использования приведенного алгоритма для расчета алгебраической иммунности векторного отображения (S-блока) необходимо перевести (отобразить) нелинейный узел в булеву функцию $f(x)$ в соответствии с (4), например, в виде полинома Жегалкина (6).

Алгоритм преобразования последовательности (4) в полином Жегалкина в общем виде сводится к решению уравнений, слагаемыми в которых, являются все возможные “составные” мономы, относительно данного и сам искомый моном, в правой части уравнения стоит значение булевой функции, относительно искомого монома. Стандартное решение может выглядеть как поиск “вхождения” каждого следующего монома в искомый, т. е. (65536×65536) операций, без учёта вспомогательных.

Для оптимизации необходимо найти зависимость искомого монома от позиций возможных мономов “вхождения”. Если рассмотреть искомый моном и сохранить позиции значения 0, то можно найти закономерность позиций, на которых точно не будут находиться мономы, “входящие” в искомый. Следовательно, пропуская априори, неподходящие нам позиции, мы можем двигаться только по позициям возможного вхождения мономов в искомый, что существенно сокращает поиск мономов “вхождения” и повышает быстродействие алгоритма преобразования к полиному Жегалкина.

Теперь дополним предыдущий алгоритм двумя новыми первыми пунктами по преобразованию S-блока в полином Жегалкина. В итоге этот модифицированный алгоритм будет выглядеть следующим образом.

Алгоритм вычислении алгебраической иммунности S-блоков

Вход: $n \in N$, S-блок $V_n \to V_n$, заданный в виде квадратной таблицы.

Выход: Значение алгебраической иммунности $AI(f)$.

Шаг 1. Вычисляем отображение S-блока (4) в виде таблицы истинности соответствующей булевой функции с удвоенным числом переменных $f_s : V_{2n} \to \{0,1\}$.

Шаг 2. Вычисляем алгебраическую нормальную форму булевой функции $f_s : V_{2n} \to \{0,1\}$. Задаём её списком одночленов (5) с ненулевыми коэффициентами a_u, $f(x) = \sum_{u \in GF(2)^n} a_u x^u$, $a_u \in GF(2)$.

Присваиваем $d = 1$.

Шаг 3. Если $A_d^n(f) = 0$ и $A_d^n(f+1) = 0$ присваиваем $d = d + 1$ и переходим к шагу 2.

Шаг 4. Если $A_d^n(f) \neq 0$ и/или $A_d^n(f+1) \neq 0$ присваиваем $AI(f) = d$ и подаем на выход алгоритма.

Как видно при построении этого алгоритма уже предприняты шаги по сокращению объёма вычислений. Сначала вычисляется аннулирующий многочлен первой степени, затем второй, третей и вычисления заканчиваются, как только найден аннулирующий многочлен минимальной степени.

Приведём здесь результаты оценки вычислительной сложности этого алгоритма при его реализации без каких-либо упрощений.

Поиск пространства аннигиляторов осуществляется путём решения системы линейных уравнений, полученных на основе группирования неизвестных коэффициентов по всем различным мономам. Рассмотрим пример, приведенный в работе [13], где $f(x)$ – полученный многочлен в виде полинома Жегалкина; $g(x)$ – искомый аннулирущий многочлен.

Пример. Для $n = 2$ и $d = 1$ имеем:

$$f(x) = a_{00} + a_{10}x_1 + a_{01}x_2 + a_{11}x_1x_2,$$

$$g(x) = b_{00} + b_{10}x_1 + b_{01}x_2.$$

После подстановки в $f(x) \cdot g(x) = 0$ получим

$$f(x) \cdot g(x) = a_{00}b_{00} + (a_{00}b_{10} + a_{10}b_{10} + a_{10}b_{00})x_1 + (a_{00}b_{01} + a_{01}b_{01} + a_{01}b_{00})x_2 + (a_{10}b_{01} + a_{01}b_{10} + a_{11}b_{00} + a_{11}b_{10} + a_{11}b_{01})x_1x_2 = 0,$$

откуда имеем систему линейных однородных уравнений:

$$\begin{cases} a_{00}b_{00} = 0, \\ a_{00}b_{10} + a_{10}b_{10} + a_{10}b_{00} = 0, \\ a_{00}b_{01} + a_{01}b_{01} + a_{01}b_{00} = 0, \\ a_{10}b_{01} + a_{01}b_{10} + a_{11}b_{00} + a_{11}b_{10} + a_{11}b_{01} = 0 \end{cases}$$

относительно неизвестных $b_{00}, b_{10}, b_{01}, \ldots$ – коэффициентов функции $g(x)$.

Тогда, например, для функции $f(x) = x_1 + x_2$ (т.е. при $a_{00} = a_{11} = 0$ и $a_{10} = a_{01} = 1$) получим систему:

$$\begin{cases} b_{10} + b_{00} = 0, \\ b_{01} + b_{00} = 0, \\ b_{01} + b_{10} = 0. \end{cases}$$

Для решения системы уравнений, например, методом Гаусса строится таблица размером $2^{2n} \times 2^{2n}$, где n – степень S-блока. Ячейки таблицы – сгруппированный результат умножения всех мономов полинома $f(x)$ на полином $g(x)$. Группировка происходит так, что строки представляются как все возможные варианты мономов (всего их 2^{2n}), а столбцы – это неизвестные коэффициенты (их также 2^{2n}). Таким образом, значение ячейки таблички может быть = 1, если коэффициент b_i присутствует в конкретном мономе и 0, если такой коэффициент отсутствует. Таблица 1 иллюстрирует этот процесс для примера $f(x) = x_1 + x_2$.

Таблица 1

Результирующая таблица сгруппированных коэффициентов

$b_{i,j} / x$	b_{00}	b_{01}	b_{10}	b_{11}
1	0	0	0	0
x_1	1	1	0	0
x_2	1	0	1	0
x_1x_2	0	1	1	0

Для байтового S-блока (n = 8), получаем таблицу размерами 65536×65536. Для хранения одной вспомогательной таблицы, которая содержит результат умножения всех возможных мономов полинома $f(x)$ на мономы полинома $g(x)$, необходимо приблизительно 8.5 Гб ОЗУ с учётом того, что каждое число в таблице может занимать 2 байта (максимально возможное значение 65535), это без учёта дополнительных вспомогательных таблиц, которые могут также понадобится в ходе вычислений.

Если говорить о временных затратах на вычисления, цифры нас также не порадуют, нам понадобятся неоднократные проходы по всей таблице (4294967296 операций на один проход) для решения систем линейных уравнений.

3. ОПТИМИЗИРОВАННЫЙ АЛГОРИТМ

Известно, что значение алгебраической иммунности S-блока, не может быть больше, чем $n/2$ [6], следовательно, для байтового S-блока значение алгебраической иммунности не может быть больше 4-х. Следовательно, при построении таблицы перемножения всех мономов полинома $f(x)$ на мономы полинома $g(x)$ мы можем брать только те мономы полинома $g(x)$, у которых степень будет меньше, либо равна $n/2$ (для байтового S-блока это четвёрка). Количество подходящих нам мономов полинома $g(x)$ в худшем случае, нетрудно посчитать по формуле

$$k = \sum_{i=1}^{n/2} C_{2n}^{i} . \qquad (8)$$

Для байтового S-блока получим:

$$k = C_{16}^{1} + C_{16}^{2} + C_{16}^{3} + C_{16}^{4} = 2516 .$$

Итого, 2516 мономов в худшем случае, вместо 65536 для байтового S-блока. Далее все вычисления при расчёте максимально возможного значения алгебраической иммунности будут происходить уже с таблицей размерами (65536×2516), а это уже 164888576 операций для прохода по всей таблице, вместо 4294967296. И около 0.3 Гб ОЗУ на ПК, вместо 8.5 для хранения такой таблицы. Возникает только одна проблема – мономы полинома никак не упорядочены по степеням, и хранятся в произвольном порядке относительно степеней, а точнее в порядке, который задаёт арифметика цифр (0...65536). Например, моном первой степени можно встретить в позициях (1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096, 8192, 16384, 32768), где позиции > 2516 не учитываются нашей оптимизацией. Для решения этой проблемы, необходимо сгруппировать полином $g(x)$ таким образом, чтобы мономы были упорядочены по степеням. Например, можно хранить позиции интересующих нас мономов отдельно и обращаться только к нужным позициям. Это позволит работать с таблицей (65536×2516), вместо (65536×65536) и во много раз сократит временные затраты на вычисления и ОЗУ на ПК.

4. ЭКСПЕРИМЕНТЫ

В качестве тестов производительности оптимизированного и стандартного решения были взяты узлы нелинейной замены шифра AES и шифра Калина (рис. 1).

S-блок AES:

```
0x63 0x7c 0x77 0x7b 0xf2 0x6b 0x6f 0xc5 0x30 0x01 0x67 0x2b 0xfe 0xd7 0xab 0x76
0xca 0x82 0xc9 0x7d 0xfa 0x59 0x47 0xf0 0xad 0xd4 0xa2 0xaf 0x9c 0xa4 0x72 0xc0
0xb7 0xfd 0x93 0x26 0x36 0x3f 0xf7 0xcc 0x34 0xa5 0xe5 0xf1 0x71 0xd8 0x31 0x15
0x04 0xc7 0x23 0xc3 0x18 0x96 0x05 0x9a 0x07 0x12 0x80 0xe2 0xeb 0x27 0xb2 0x75
0x09 0x83 0x2c 0x1a 0x1b 0x6e 0x5a 0xa0 0x52 0x3b 0xd6 0xb3 0x29 0xe3 0x2f 0x84
0x53 0xd1 0x00 0xed 0x20 0xfc 0xb1 0x5b 0x6a 0xcb 0xbe 0x39 0x4a 0x4c 0x58 0xcf
0xd0 0xef 0xaa 0xfb 0x43 0x4d 0x33 0x85 0x45 0xf9 0x02 0x7f 0x50 0x3c 0x9f 0xa8
0x51 0xa3 0x40 0x8f 0x92 0x9d 0x38 0xf5 0xbc 0xb6 0xda 0x21 0x10 0xff 0xf3 0xd2
0xcd 0x0c 0x13 0xec 0x5f 0x97 0x44 0x17 0xc4 0xa7 0x7e 0x3d 0x64 0x5d 0x19 0x73
0x60 0x81 0x4f 0xdc 0x22 0x2a 0x90 0x88 0x46 0xee 0xb8 0x14 0xde 0x5e 0x0b 0xdb
0xe0 0x32 0x3a 0x0a 0x49 0x06 0x24 0x5c 0xc2 0xd3 0xac 0x62 0x91 0x95 0xe4 0x79
0xe7 0xc8 0x37 0x6d 0x8d 0xd5 0x4e 0xa9 0x6c 0x56 0xf4 0xea 0x65 0x7a 0xae 0x08
0xba 0x78 0x25 0x2e 0x1c 0xa6 0xb4 0xc6 0xe8 0xdd 0x74 0x1f 0x4b 0xbd 0x8b 0x8a
0x70 0x3e 0xb5 0x66 0x48 0x03 0xf6 0x0e 0x61 0x35 0x57 0xb9 0x86 0xc1 0x1d 0x9e
0xe1 0xf8 0x98 0x11 0x69 0xd9 0x8e 0x94 0x9b 0x1e 0x87 0xe9 0xce 0x55 0x28 0xdf
0x8c 0xa1 0x89 0x0d 0xbf 0xe6 0x42 0x68 0x41 0x99 0x2d 0x0f 0xb0 0x54 0xbb 0x16
```

S-блок Калина:

```
0xA8 0x43 0x5F 0x06 0x6B 0x75 0x6C 0x59 0x71 0xDF 0x87 0x95 0x17 0xF0 0xD8 0x09
0x6D 0xF3 0x1D 0xCB 0xC9 0x4D 0x2C 0xAF 0x79 0xE0 0x97 0xFD 0x6F 0x4B 0x45 0x39
0x3E 0xDD 0xA3 0x4F 0xB4 0xB6 0x9A 0x0E 0x1F 0xBF 0x15 0xE1 0x49 0xD2 0x93 0xC6
0x92 0x72 0x9E 0x61 0xD1 0x63 0xFA 0xEE 0xF4 0x19 0xD5 0xAD 0x58 0xA4 0xBB 0xA1
0xDC 0xF2 0x83 0x37 0x42 0xE4 0x7A 0x32 0x9C 0xCC 0xAB 0x4A 0x8F 0x6E 0x04 0x27
0x2E 0xE7 0xE2 0x5A 0x96 0x16 0x23 0x2B 0xC2 0x65 0x66 0x0F 0xBC 0xA9 0x47 0x41
0x34 0x48 0xFC 0xB7 0x6A 0x88 0xA5 0x53 0x86 0xF9 0x5B 0xDB 0x38 0x7B 0xC3 0x1E
0x22 0x33 0x24 0x28 0x36 0xC7 0xB2 0x3B 0x8E 0x77 0xBA 0xF5 0x14 0x9F 0x08 0x55
0x9B 0x4C 0xFE 0x60 0x5C 0xDA 0x18 0x46 0xCD 0x7D 0x21 0xB0 0x3F 0x1B 0x89 0xFF
0xEB 0x84 0x69 0x3A 0x9D 0xD7 0xD3 0x70 0x67 0x40 0xB5 0xDE 0x5D 0x30 0x91 0xB1
0x78 0x11 0x01 0xE5 0x00 0x68 0x98 0xA0 0xC5 0x02 0xA6 0x74 0x2D 0x0B 0xA2 0x76
0xB3 0xBE 0xCE 0xBD 0xAE 0xE9 0x8A 0x31 0x1C 0xEC 0xF1 0x99 0x94 0xAA 0xF6 0x26
0x2F 0xEF 0xE8 0x8C 0x35 0x03 0xD4 0x7F 0xFB 0x05 0xC1 0x5E 0x90 0x20 0x3D 0x82
0xF7 0xEA 0x0A 0x0D 0x7E 0xF8 0x50 0x1A 0xC4 0x07 0x57 0xB8 0x3C 0x62 0xE3 0xC8
0xAC 0x52 0x64 0x10 0xD0 0xD9 0x13 0x0C 0x12 0x29 0x51 0xB9 0xCF 0xD6 0x73 0x8D
0x81 0x54 0xC0 0xED 0x4E 0x44 0xA7 0x2A 0x85 0x25 0xE6 0xCA 0x7C 0x8B 0x56 0x80
```

Рис. 1.

При построении полинома Жегалкина с помощью стандартного алгоритма реализации необходимо было около 20 минут. Оптимизированный алгоритм построения полинома Жегалкина справился с поставленной задачей менее чем за 2 секунды. Замеры производились на ПК с Windows 10, Intel Core i7-3630QM 2.4 Ггц.

Алгебраическая иммунность S-блока шифра AES равна 2. Воспользовавшись выражением (8), имеем: $k = C_{16}^1 + + C_{16}^2 = 136$. Алгебраическая иммунность S-блока шифра Калина равна 3, т.е., имеем: $k = C_{16}^1 + C_{16}^2 + C_{16}^3 = 696$. Именно такие значения получены с помощью системы «Magma» в работе [13].

С помощью оптимизированного алгоритма заведомо выбирая только необходимые коэффициенты для подсчёта алгебраической иммунности шифров AES и Калина, получаем таблицы предвычислений размерами (65536×136) и (65536×696) соответственно, вместо (65536×65536).

Время выполнения для стандартного алгоритма выходит за приемлемые сроки. Подсчёт алгебраической иммунности одного узла нелинейной замены занимал бы более одной недели. Оптимизированный алгоритм вычислений выполнил поставленную задачу за 4 секунды для S-блока AES и за 20 с для S-блока Калина-2 соответственно. Замеры производились на ПК с Windows 10, Intel Core i7-3630QM 2.4 Ггц.

5.ОБСУЖДЕНИЕ

Предложен ускоренный алгоритм расчёта алгебраической иммунности, байтовых S-блоков, в котором используются реально существующие практические и теоретические ограничения, характерные для оцениваемого показателя. Он позволяет существенно сократить объёмы обрабатываемой информации. В качестве таких ограничений использовано то, что в соответствии с теорией алгебраическая иммунность для байтовых S-блоков не может превышать значения 4, т.е. степень аннулирующего многочлена не может быть выше четвёртой. Последующая операция приведения сокращённой по числу столбцов матрицы коэффициентов к диагональному виду позволяет сократить размеры матрицы коэффициентов и по числу строк (появляются строки из одних нулей). В результате действительно удаётся существенно ускорить процедуру выполнения вычислений.

ВЫВОДЫ

Таким образом, основным результатом работы является обоснование ускоренного метода расчёта алгебраической иммунности S-блоков. Исключение при формировании программы множеств данных, не участвующих на каждом из этапов её работы в формировании результата, а также учёт априорно известных данных относительно конечного результата позволили существенно сократить объёмы промежуточных вычислений и добиться повышения производительности программы в сотни раз. Время работы программы при расчёте алгебраической иммунности S-блока со значением этого показателя равного трём составляет около 20-ти с.

Литература

[1] *A.J. Menezes, P.C. van Oorschot, S.A. Vanstone*. Handbook of Applied Cryptography. CRC Press, 1997, 794 p.

[2] *N. Ferguson and B. Schneier.* Practical Cryptography. John Wiley & Sons, 2003, 432 p.

[3] *Горбенко І.Д., Горбенко Ю.І.* Прикладна криптологія. Теорія. Практика. Застосування: монографія.– Харків: Видавництво «Форт», 2012. – 870 с.

[4] ISCI'2018: Information Security in Critical Infrastructures. Collective monograph. Edited by Ivan D. Gorbenko and Alexandr A. Kuznetsov, ASC Academic Publishing, USA, 2018, 360 p.

[5] *Courtois N. and Meier W.* Algebraic attacks on stream ciphers with linear feedback // Eurocrypt'2003. LNCS. 2003. V. 2656. P. 345–359.

[6] *Покрасенко Д.П.* Об алгебраической иммунности векторных булевых функций / Д.П. Покрасенко // Прикладная дискретная математика. Приложение, 2014, № 7, 43–48.

[7] *Meier W., Pasalic E., and Carlet C.* Algebraic attacks and decomposition of Boolean functions // Eurocrypt'2004. LNCS. 2004. V. 3027. P. 474–491. 3. Armknecht F. and Krause M. Constructing single- and multi-output Boolean functions with maximal immunity // ICALP'2006. LNCS. 2006. V. 4052. P. 180–191.

[8] *Armknecht F. and Krause H.* Constructing single- and multi-output Boolean functions with maximal immunity // ICALP'2006/ V/4052. P. 180–191.

[9] *Ars G. and Faug`ere J.-C.* Algebraic immunities of functions over finite fields // Proc. Conf. BFCA. 2005. P. 21–38.

[10] *Carlet C.* On the algebraic immunities and higher order nonlinearities of vectorial Boolean functions // Enhancing Cryptographic Primitives with Techniques from Error Correcting Codes. Amsterdam: IOS Press, 2009. P. 104–116.

[11] *Faugère, J.-C.* (June 1999). A new efficient algorithm for computing Gröbner bases (F4). Journal of Pure and Applied Algebra. Elsevier Science. 139 (1): 61–88.

[12] *Покрасенко Д.П.* Компонентная алгебраическая иммунность S-блоков, использующихся в некоторых блочных шифрах // Прикладная дискретная математика. Приложение, 2017, № 10, 49–51.

[13] *Кузнецов О.О.* Алгебраїчний імунітет нелінійних вузлів симетричних шифрів / О.О. Кузнецов, Ю.І. Горбенко, І.М. Білозерцев та інші // Радиотехника. – 2017. – Вып. 189. –С. 47–58.

[14] Magma Computational Algebra System. Available at: http://magma.maths.usyd.edu.au/magma.

[15] *Massimiliano Sala, Teo Mora, Ludovic Perret, Shojiro Sakata, Carlo Traverso Gröbner* Bases, Coding, and Cryptography. Springer-Verlag Berlin Heidelberg. – 426 p.

[16] *Баев Владимир Валерьевич*. Эффективные алгоритмы получения оценок алгебраической иммунности булевых функций: диссертация на соискание ученой степени кандидата физико-математических наук: 01.01.09 – Москва, 2008. – 101 с.

[17] *Аржанцев И.В.* Базисы Грёбнера и системы алгебраических уравнений. Летняя школа. Современная математика. Дубна, июль 2002. – Москва: МЦНМО, 2003. – 68 с.

Поступила в редколлегию 20.11.2018

Лисицкий Константин Евгеньевич, аспирант кафедры безопасности информационных систем и технологий Харьковского национального университета им. В.Н. Каразина. Область научных интересов – криптография, технологии блочного симметричного шифрования.

Кузнецов Александр Александрович, доктор технических наук, профессор, заместитель главного конструктора АТ «ИИТ», профессор кафедры безопасности информационных систем и технологий Харьковского национального университета им. В.Н. Каразина. Область научных интересов – криптография и аутентификация, алгебраическая теория кодов, обработка, передача и защита информации.

Горбенко Юрий Иванович, кандидат технических наук, исполнительный директор АТ «ИИТ», старший научный сотрудник кафедры Харьковского национального университета имени В.Н. Каразина. Область научных интересов – криптография и аутентификация, инфраструктура открытых ключей.

Оноприенко Виктор Васильевич, кандидат технических наук, доцент, генеральный директор АТ «Институт информационных технологий». Область научных интересов – криптография и аутентификация, инфраструктура открытых ключей, теория защиты информации, информационная и кибербезопасность государства.

Стельник Игорь Валерьевич, заместитель директора департамента защиты информации Администрации Государственной службы специальной связи и защиты информации Украины. Область научных интересов – криптография и аутентификация, теория защиты информации.

УДК 004.056.55

Лисицький К.Є. **Прискорений метод реалізації обчислення алгебраїчної імунності нелінійних вузлів заміни блокових симетричних шифрів** / К.Є. Лисицький, О.О. Кузнецов, Ю.І. Горбенко, В.В. Оноприєнко, І.В. Стельник // Прикладна радіоелектроніка: наук.–техн. журнал. – 2018. –Том 17. № 3, 4. – С. 81–87.

Розглядається прискорений метод обчислення алгебраїчної імунності нелінійних вузлів заміни симетричних шифрів по Жан-Шарлю Фожеру (Jean-Charles Faugère). Він заснований на пошуку анігілюючої функції до полінома Жегалкина, побудованої з вихідного нелінійного вузла заміни. Проводиться аналіз швидкодії відомого алгоритму підрахунку алгебраїчної імунності. Обговорюються деталі реалізації алгоритму і надається опис покращеного алгоритму обчислення алгебраїчної імунності, оптимізованого за часом і обсягом затрат і ресурсів.

Ключові слова: симетричний шифр, алгебраїчний імунітет, нелінійний вузол заміни, булева функція, продуктивність алгоритму.

Табл.: 01. Іл.: 01. Бібліогр.: 17 найм.

UDC 004.056.55

Lisitsky K. **An accelerated method of calculating the algebraic immunity of nonlinear nodes of replacing symmetric ciphers** / K. Lisitsky, A. Kuznetsov, Yu. Gorbenko, V.Onopryenko, I.Stelnik // Applied Radio Electronics: Sci. Journ. – 2018. – Vol. 17, № 3, 4. – P. 81–87.

An accelerated method for calculating the algebraic immunity of nonlinear replacement nodes of symmetric ciphers by Jean-Charles Faugère is considered. It is based on the search for the annihilating function to the Zhegalkin polynomial constructed from the original nonlinear node of substitution. An analysis of the speed of a known algorithm for calculating algebraic immunity is given. The details of the implementation of the algorithm are discussed and a description of the accelerated algorithm for calculating algebraic immunity, optimized in terms of computation time and for the resources of the RAM is provided.

Keywords: symmetric cipher, algebraic immunity, non-linear replacement node, Boolean function, algorithm performance.

Tab. 01. Fig. 01. Ref.: 17 items.