Алгоритм (12) – (14) допускает дальнейшие модификации, улучшающие его стойкость, в частности применение системы омофонов для маскировки частот появления коэффициентов V_i, $i=\overline{1,I}$. Для скрытия вероятностных связей более высоких порядков могут быть использованы соответствующие канонические разложения [8 – 9].

III Выводы

Анализ стохастических методов шифрования данных показал, что существующие алгоритмы не позволяют полностью скрыть вероятностные связи исходного сообщения и, таким образом, совершенствование данных методов является актуальной проблемой.

В работе рассмотрен метод шифрования, который позволяет преобразовать передаваемое сообщение в последовательность некоррелированных значений, что существенно затрудняет задачу вскрытия исходных данных. Алгоритм базируется на каноническом разложении исследуемой случайной последовательности.

Предложенный метод шифрования проверен для украинского языка. В качестве исходных данных для определения параметров алгоритма был использован украинский толковый словарь (22 тыс. слов).

В работе также предложены дальнейшие пути совершенствования полученного алгоритма шифрования.

Литература: ***1.*** *Іванов М. А., Чугункою И. В. Теория, применение и оценка качества генераторов псевдослучайных последовательностей.- М.: КУДИЦ-ОБРАЗ, 2003. – 240 с.* ***2.*** *Іванов М. А. Криптографические методы защиты информации в компьютерных системах и сетях. - М.: КУДИЦ-ОБРАЗ, 2001. – 361 с.* ***3.*** *Alfred Menezes., Minghua Qu., Scott Vanstone. IEEE P1363, Part 4: Elliptic Curve Systems. 1995.* ***4.*** *Gong G., Lam C.C.Y. Linear Recursive Sequences over Elliptic Curves. 2001.http:citeseer.ist.psu.edu/449444. html.* ***5.*** *Кулаков И. А. Стохастические системы и их применение в криптографии. Дихотомические последовательности и генераторы. – Материалы 8-ой конференции «РусКрипто'2006».* ***6.*** *Введение в криптографию / Под общ. ред. В. В. Ященко. М.: МЦНМО, 2000.* ***7.*** *Пугачев В. С. Теория случайных функций и ее применение. - М.: Физматгиз, 1962. - 720 с.* ***8.*** *Атаманюк И. П. Полиномиальный алгоритм оптимальной экстраполяции параметров стохастических систем. //Управляющие системы и машины. – 2002. - №1.* ***9.*** *Атаманюк И. П. Алгоритм реализации нелинейной случайной последовательности на базе ее канонического разложения. //Электронное моделирование. - 2001. -№5. с. 38 - 46.*

УДК 621.391:519.7:510.5

АЛГОРИТМ ФОРМИРОВАНИЯ МАТРИЦ НАД ПРИМАРНЫМ КОЛЬЦОМ ВЫЧЕТОВ ДЛЯ ПОСТРОЕНИЯ ПРОТОКОЛОВ МНОЖЕСТВЕННОГО РАЗДЕЛЕНИЯ СЕКРЕТА, РЕАЛИЗУЮЩИХ ЗАДАННУЮ ИЕРАРХИЮ ДОСТУПА

Андрей Волошин

Институт специальной связи и защиты информации НТУУ "КПИ"

Аннотация: **Предложен алгоритм формирования матриц над примарным кольцом вычетов, предназначенных для построения линейных совершенных протоколов множественного разделения секрета для заданной иерархии доступа. Указанный алгоритм обобщает известный ранее алгоритм формирования матриц над конечным полем для синтеза линейных протоколов разделения одного секрета и имеет меньшую временную сложность по сравнению с тривиальным алгоритмом.**

Summary: **Perfect linear multi-secret sharing schemes over primary residue ring construction algorithm is proposed. Early known secret sharing schemes over finite field construction method is generalized by proposed algorithm. This algorithm has calculation complexity, which less compare with trivial algorithm.**

Ключевые слова: **Криптографическая защита информации, протокол множественного разделения секрета, иерархия доступа, кольцо вычетов.**

I Введение

Протокол или схема разделения секрета (ПРС) представляет собой криптографический протокол, позволяющий "разделить" некоторый секретный параметр (секрет) среди множества участников протокола таким образом, чтобы только некоторые, заранее определенные (разрешенные) коалиции участников могли восстановить его значение при объединении хранящейся у них индивидуальной секретной информации (проекций секрета). Протокол разделения секрета, в котором участники запрещенных коалиций не могут

получить никакой информации о значении секрета, называется совершенным [1]. Применение ПРС при построении подсистем управления доступом современных информационно-телекоммуникационных систем позволяет, как правило, повысить уровень защищенности их информационных ресурсов [2, 3].

Свойства и способы построения протоколов разделения единственного секрета известны с 1979 года [4, 5] и в дальнейшем интенсивно изучались в работах [6 – 12] и ряде других. В силу простоты схемно-технической реализации и вычислительной эффективности особый интерес исследователей вызвали конструкции линейных ПРС, основанных на линейных (над конечными полями, кольцами вычетов и т. д.) математических преобразованиях (см., например, работы [6, 7]). Так, в [11] предложена конструкция линейного (над кольцом Галуа) несовершенного ПРС, а в [12] – "обобщенная векторная конструкция" линейных совершенных протоколов разделения секрета (над прямым произведением векторных пространств).

Естественным обобщением ПРС на случай нескольких секретов являются протоколы множественного разделения секрета (ПМРС), впервые введенные в [13] и формально описанные в [14]. Такие протоколы позволяют решать более разнообразный, по сравнению с ПРС, спектр задач, связанных с разграничением доступа к ресурсам ИТС, и имеют более широкую сферу практического применения [13].

В [15, 16] предложен метод построения линейных совершенных протоколов множественного разделения секрета с использованием матриц над кольцом вычетов целых чисел и исследованы свойства таких ПМРС. В статье [17] получено аналитическое описание конструкций указанных ПМРС, соответствующих матрицам над примарными кольцами вычетов и реализующих заранее определенные иерархии доступа. Для проверки существования и нахождения в явном виде матриц, задающих линейные ПМРС над примарным кольцом вычетов, в соответствии с результатами [15, 17] можно использовать тривиальный (переборный) алгоритм, однако на практике такой алгоритм, как правило, является неэффективным.

Данная статья посвящена разработке более эффективного, по сравнению с тривиальным, алгоритма формирования матриц над примарным кольцом вычетов, необходимых для синтеза линейных совершенных ПМРС, реализующего заданную совокупность множеств.

II Основные понятия, обозначения и вспомогательные результаты

Приведем результаты, изложенные в [15, 17], которые используются в дальнейшем изложении.

Пусть M – подмодуль свободного модуля размерности $n + 1$ над кольцом $R = Z/p^d$ (где p – простое число, $d \geq 1$), порожденный строками $k \times (n + 1)$-матрицы

$$G = \left(\begin{array}{c|c} 1 & \\ 0 & \\ \vdots & G' \\ 0 & \end{array}\right), \qquad (1)$$

где G' – матрица размера $k \times n$ над кольцом R, $k \geq 1$. Согласно [15], модулю M соответствует ПМРС $\sigma(G)$ на множестве участников $P = \{1, 2, \ldots, n\}$. Протокол множественного разделения секрета $\sigma(G)$ осуществляет разделение множества секретов $S_0 = \{(s_l)\colon s_l \in \mathrm{GF}(p),\ l \in \overline{0, d-1}\}$. Обозначим $\tilde{\Sigma}_i$, $i \in \overline{0,d}$ совокупность множеств A участников ПМРС $\sigma(G)$, которые способны восстановить секреты s_l с номерами $0 \leq l \leq d - i$. Совокупность множеств $\tilde{\Sigma} = \left\{\tilde{\Sigma}_i : i \in \overline{0,d}\right\}$ называется иерархией доступа ПМРС $\sigma(G)$ [15].

Для любого $A \in 2^P$ обозначим $\left(\vec{n}\right)_A$ вектор, который состоит из координат вектора $\vec{n}$, номера которых принадлежат множеству A, и G_A – матрицу, которая состоит из столбцов матрицы G, номера которых принадлежат множеству A. Пусть также $\left\langle G_A \right\rangle_R$ – R-модуль, порожденный столбцами матрицы G_A. В [15] показано, что $A \in \tilde{\Sigma}_i$ в том и только в том случае, когда i является наименьшим целым числом от 0 до d, удовлетворяющим условию

$$p^i G_0 \in \left\langle G_A \right\rangle_R. \qquad (2)$$

Пусть теперь $\Sigma = \left\{\Sigma_i : i \in \overline{0,d}\right\}$ – совокупность попарно непересекающихся подмножеств множества 2^P (случай $\Sigma_i = \varnothing$ не исключается). Необходимо разработать алгоритм формирования матрицы G вида (1), задающий ПМРС указанного выше типа, который реализует совокупность множеств Σ в качестве иерархии

доступа $\widetilde{\Sigma}$. Такой алгоритм должен определять, существует ли для совокупности множеств Σ матрица G вида (1), удовлетворяющая условиям (2) для всех $A \in \Sigma_i, i \in \overline{0,d}$, и, в случае существования, строить ее в явном виде.

Рассмотрим классы $\Delta_i = \bigcup_{j=0}^{i} \Sigma_j$, $i \in \overline{0,d}$. Обозначим $\Sigma_i^0 = \{A_{i,1}, \ldots, A_{i,r_i}\}$ и $\Delta_i^0 = \{B_{i,1}, \ldots, B_{i,q_i}\}$ множества минимальных (относительно включения) элементов классов Σ_i и Δ_i соответственно; для любого $X \subset P$, $\Delta_i \subset 2^P$, $i \in \overline{0,d}$ положим $\tilde{O} = Ð \setminus \tilde{O}$, $\overline{\Delta}_i = 2^P \setminus \Delta_i$ и обозначим $\overline{\Delta}_i^1$ множество максимальных (относительно включения) элементов класса $\overline{\Delta}_i$, $i \in \overline{0,d-1}$.

Пусть C_Σ – некоторая матрица над кольцом R, состоящая из $r = r_0 + \ldots + r_{d-1}$ строк $\overrightarrow{c_{i,j}} \in R^{n+1}$, где $i \in \overline{0,d-1}$, $j \in \overline{1,r_i}$. Будем говорить, что матрица C_Σ удовлетворяет условию (*), если для любых $i \in \overline{0,d-1}$, $j \in \overline{1,r_i}$ выполняется равенство $\overrightarrow{c_{i,j}} = \left(p^i, \overrightarrow{c'_{i,j}}\right)$, где носитель $\mathrm{supp}(\overrightarrow{c'_{i,j}})$ (множество номеров ненулевых координат вектора $\overrightarrow{c'_{i,j}}$) равен $A_{i,j}$.

Анализируя условие (2), можно предложить тривиальный (переборный) алгоритм формирования искомой матрицы G для заданной совокупности множеств Σ. Он заключается в переборе всех возможных матриц G', в общем случае, размера $(n + 1) \times n$ над кольцом R и проверке выполнения условия (2) для всех $A \in \Sigma_i^0$, $i \in \overline{0,d}$. Оценка трудоемкости такого алгоритма приведена в разделе IV.

Отметим, что, согласно [17], для существования матрицы, которая определяет ПМРС, реализующий совокупность Σ в качестве иерархии доступа $\widetilde{\Sigma}$, необходимо выполнение следующих условий:

(а) $\varnothing \in \Sigma_d$;

(б) для любых $i \in \overline{0,d}$ класс Δ_i является монотонным.

В [17] получено аналитическое описание конструкции ПМРС над кольцом R, реализующего совокупность множеств Σ в качестве иерархии доступа $\widetilde{\Sigma}$; в частности, показано, что матрица G вида (1) над кольцом R удовлетворяет условиям (2) для всех $A \in \widetilde{\Sigma}_i$, $i \in \overline{0,d}$, тогда и только тогда, когда матрица C, ортогональная к G ($GC^{\mathrm{T}} = 0$), обладает такими свойствами:

- C удовлетворяет условию (*);

- для любых $l \in \overline{0,d-1}$, $\tilde{O} \in \overline{\Delta}_l^1$, совместна система линейных уравнений (СЛУ)

$$C_{\overline{X}} \tilde{o}^{\downarrow} = p^{d-(l+1)} c_0^{\downarrow} \qquad (3)$$

над кольцом R.

Нетрудно видеть, что при проверке условия (3) матрица C_Σ с самого начала может быть выбрана таким образом, чтобы в каждом ее столбце первый сверху ненулевой элемент равнялся некоторой степени числа p. Действительно, умножение каждого столбца матрицы C_Σ на произвольный обратимый элемент кольца R не нарушает совместности СЛУ (3) для всех $l \in \overline{0,d-1}$, $\tilde{O} \in \overline{\Delta}_l^1$.

Отметим, что если матрица C, удовлетворяющая условию (3), задана, то по ней нетрудно непосредственно построить матрицу G вида (1), которая определяет ПМРС $\sigma(G)$. А именно, согласно [17], в силу ортогональности этих матриц в качестве матрицы G может быть выбрана система образующих модуля решений СЛУ $Cx^{\downarrow} = 0^{\downarrow}$ над кольцом R.

Как будет показано дальше, совокупность Σ однозначно определяется совокупностью множеств Δ_i^0, $i \in \overline{0,d-1}$. Исходя из этого, будем рассматривать совокупность множеств Δ_i^0, $i \in \overline{0,d-1}$ одними из входных данных разрабатываемого алгоритма.

Далее в этой статье излагается и обосновывается алгоритм, который по данной совокупности множеств

$\Delta_i^0,\ i \in \overline{0,d-1}$ позволяет проверить существование матрицы *G* вида (1), удовлетворяющей условиям (2) для всех $A \in \Sigma_i, i \in \overline{0,d}$, и, в случае ее существования, построить в явном виде матрицу C_Σ, обладающую приведенными свойствами.

Дальнейшее изложение построено таким образом. Описанию алгоритма посвящено содержание раздела II. Раздел III содержит результаты анализа временной и емкостной сложностей вычислительных процедур предложенного алгоритма и, наконец, в разделе IV приводится пример его практического применения.

III Описание и математическое обоснование алгоритма формирования матриц для построения ПМРС, реализующих заданную иерархию доступа

Алгоритм предназначен для формирования по заданной совокупности множеств матрицы над примарным кольцом вычетов, которая задает протокол множественного разделения секрета, реализующий эту совокупность в качестве иерархии доступа.

Исходными данными для формирования матрицы над кольцом *R* в соответствии с предлагаемым алгоритмом являются:

1) число *n* участников ПМРС;
2) простое число *p*, целое число $d \geq 1$;
3) совокупности Δ_i^0 попарно не содержащих друг друга подмножеств множества $P = \{1, 2, \ldots, n\}$ $i \in \overline{0,d-1}.$

Допущения и ограничения. Моделью вычислительного устройства, используемого для реализации предлагаемого алгоритма, является равнодоступная адресная машина (РАМ) [18]. Под элементарной операцией (ЭО) понимается арифметическая операция (сложения, вычитания, умножения, обращения) в кольце *R*.

Алгоритм состоит из трех этапов, выполняемых последовательно. На первом этапе (предварительных вычислений) осуществляется проверка монотонности классов $\Delta_i,\ i \in \overline{0,d-1}$ и построение совокупностей множеств Σ_i^0 и $\overline{\Delta}_i^1,\ i \in \overline{0,d-1}$. На втором этапе (построения промежуточной матрицы) осуществляется формирование матрицы C_Σ по методу поиска с возвращением. На третьем этапе (построения результирующей матрицы) выбирается система образующих модуля решений СЛУ $C_\Sigma x^{\downarrow} = 0^{\downarrow}$ вида (1) над кольцом *R*.

Этап предварительных вычислений.

Входными данными для процедур, выполняемых на этапе предварительных вычислений, является совокупность множеств $\Delta_i^0,\ i \in \overline{0,d-1}$. На этом этапе выполняются следующие процедуры.

1. Проверка монотонности классов $\Delta_i,\ i \in \overline{0,d-1}.$
2. Построение по заданным классам Δ_i^0 совокупности множеств $\Sigma_i^0,\ i \in \overline{0,d-1}.$
3. Построение по заданным классам Δ_i^0 совокупности множеств $\overline{\Delta}_i^1,\ i \in \overline{0,d-1}.$

Приведем детальное описание указанных процедур.

1. Как известно [1], монотонность класса $\Delta_i,\ i \in \overline{0,d-1}$, равносильна условию

$$B_{i,s} \not\subset B_{i,t},\ s \neq t,\ s,t \in \overline{1,q_i}. \tag{4}$$

Итак, для проверки (4) необходимо выполнить $\binom{\left|\Delta_i^0\right|}{2},\ i \in \overline{0,d-1}$ операций проверки включения множеств друг в друга.

2. Для построения совокупности множеств $\Sigma_i^0,\ i \in \overline{0,d-1}$ воспользуемся следующим утверждением:

$$\Sigma_0^0 = \Delta_0^0, \tag{5}$$

$$\Sigma_i^0 = \Delta_i^0 \setminus \Delta_{i-1} \text{ , для всех } i \in \overline{1, d-1}. \quad (6)$$

Справедливость равенства (5) вытекает из определения множеств Δ_0^0. Для доказательства равенства (6) отметим, что, согласно [17], справедливы соотношения $\Sigma_i^0 = \Delta_i^0 \cap \Sigma_i$, $i \in \overline{0, d-1}$ из которых, в силу определения множеств Δ_i^0, получаем, что

$$\Sigma_i^0 = \Delta_i^0 \cap \Sigma_i = \Delta_i^0 \cap (\Delta_i \setminus \Delta_{i-1}) = \Delta_i^0 \setminus \Delta_{i-1},$$

что и требовалось доказать.

Равенства (5), (6) позволяют предложить следующий алгоритм построения множеств Σ_i^0, $i \in \overline{1, d-1}$. Для данного $i \in \overline{1, d-1}$ и каждого $\nu \in \overline{1, q_i}$ проверяется условие

$$\hat{A}_{i,\nu} \not\supseteq \hat{A}_{i-1,\mu} \text{ , } \mu \in \overline{1, q_{i-1}}. \quad (7)$$

Если условие (7) выполняется, то $B_{i,\nu} \in \Sigma_i^0$, в противном случае – $B_{i,\nu} \notin \Sigma_i^0$.

Временная сложность построения совокупности множеств Σ_i^0, $i \in \overline{1, d-1}$, составляет $\sum_{i=1}^{d-1} q_i q_{i-1}$ операций проверки включения множеств друг в друга.

3. Для построения совокупности множеств $\overline{\Delta}_i^1$, $i \in \overline{0, d-1}$, зафиксируем число $i \in \overline{0, d-1}$, обозначим $M_1, \ldots, M_s$ все минимальные по включению подмножества множества P такие, что $M_k \cap B_j \neq \varnothing$, для всех $k \in \overline{1, s}$, $j \in \overline{1, q_i}$ (тут и далее индекс i опущен). Покажем, что

$$\overline{\Delta}_i^1 = \left\{\overline{M_1}, \ldots, \overline{M_s}\right\}. \quad (8)$$

Для доказательства равенства (8) отметим, прежде всего, что $\overline{M_k} \in \overline{\Delta}_i$, $k \in \overline{1, s}$. Это включение справедливо в силу монотонности класса $\overline{\Delta}_i$ и условия $P \setminus M_k \not\supseteq B_j$ для каждого $j \in \overline{1, q_i}$. Действительно, в противном случае $M_k \cap B_j = \varnothing$, что, однако, противоречит определению множеств M_k, $k \in \overline{1, s}$.

Далее, докажем включение $\overline{\Delta}_i^1 \subseteq \left\{\overline{M_1}, \ldots, \overline{M_s}\right\}$. Пусть $D \in \overline{\Delta}_i^1$ – максимальный элемент множества $\overline{\Delta}_i$ и $a_j \in B_j \setminus D$, $j \in \overline{1, q_i}$. Рассмотрим множество $M = \{a_1, \ldots, a_{q_i}\}$. Из условия $P \setminus M \not\supseteq B_j$, $j \in \overline{1, q_i}$ и минимальности множеств $\overline{M_k}$, $k \in \overline{1, s}$ вытекает, что множество M содержит некоторое подмножество M_k. Итак, справедливы включения $D \subseteq P \setminus M \subseteq P \setminus M_k$. Таким образом, все максимальные элементы множества $\overline{\Delta}_i$ содержатся в множестве $\left\{\overline{M_1}, \ldots, \overline{M_s}\right\}$.

Отметим, наконец, что $\overline{M_k} \in \overline{\Delta}_i^1$ для любого $k \in \overline{1, s}$, потому что $\overline{\Delta}_i^1$ – антимонотонный класс, и множества $\overline{M_k}$, $k \in \overline{1, s}$, в силу их определения, попарно не содержат одно другого.

Итак, равенство (8) полностью доказано.

Приведем алгоритм построения совокупности множеств $\overline{\Delta}_i^1$, $i \in \overline{0, d-1}$, основанный на формуле (8).

Обозначим H матрицу инцидентности совокупности множеств $\overline{\Delta}_i^1$. Далее, переберем подмножества A множества P, которые состоят из n, $n-1$, $n-2$, …, 1 элементов, проверяя для каждого из них следующие условия:

a) в подматрице H_A все строки ненулевые;

b) удаление любого столбца из H_A приводит к появлению в полученной матрице хотя бы одной нулевой строки.

Подмножества множества P, удовлетворяющие условиям a) и b), и будут искомыми $M_1, \ldots, M_s$.

Временная сложность построения совокупности множеств $\overline{\Delta}_i^1$, $i \in \overline{0, d-1}$, в наихудшем случае

составляет $2^n nd$ операций сравнения элемента матрицы H с нулем.

Этап построения промежуточной матрицы C_Σ.

Входными данными для вычислительной процедуры, выполняемой на этапе построения матрицы C_Σ, являются совокупности множеств Σ_i^0 и $\overline{\Delta}_i^1$, $i \in \overline{0, d-1}$.

Этот этап заключается в последовательном формировании строк указанной матрицы и проверки выполнения для текущей матрицы критерия, изложенного в разделе I. Процедура завершает работу, если в результате ее выполнения получена матрица C_Σ, которая состоит из r строк и удовлетворяет указанному критерию, или сделан вывод о несуществовании такой матрицы для заданных входных данных.

Порядок следования индексов i, j при формировании строк $\overrightarrow{c_{i,j}}$ матрицы C_Σ такой:

1) в направлении возрастания номеров множеств Σ_i^0 (от 0 до d–1);

2) в каждом множестве Σ_i^0 – в направлении возрастания номеров его элементов (от 1 до r_i).

Обозначим m номер текущей строки в матрице C.

Процедура построения матрицы C_Σ состоит из последовательности одинаковых по содержанию шагов, каждый из которых заключается в выборе вектора $\overrightarrow{c_{i,j}}$, $i \in \overline{0, d-1}$, $j \in \overline{1, r_i}$ – «кандидата» на очередную (m-ую) строку матрицы C и проверке условия совместности соответствующих СЛУ вида (3) для всех $l \in \overline{0, d-1}$, $\tilde{O} \in \overline{\Delta}_l^1$. Если все указанные СЛУ совместны, то выбирается следующая строка матрицы C и т.д. В случае, если для текущего вектора $\overrightarrow{c_{i,j}}$, $i \in \overline{0, d-1}$, $j \in \overline{1, r_i}$, несовместна хотя бы одна СЛУ (для некоторого $\tilde{O} \in \overline{\Delta}_l^1$, $l \in \overline{0, d-1}$), то выбирается другой вектор $\overrightarrow{c_{i,j}}$ (для тех же значений i и j) и условие совместности СЛУ проверяется уже для него. Если перебраны все возможные значения вектора $\overrightarrow{c_{i,j}}$, получено заключение о несовместности СЛУ и $m \neq 1$, положить $m = m - 1$ и изменить предыдущую (m - 1-ую) строку матрицы C. Если при этом m = 1, сделать вывод о том, что для заданных входных данных (а значит, и для совокупности множеств Σ_i, заданной совокупностью множеств Δ_i^0, $i \in \overline{0, d-1}$) матрица C_Σ не существует. Процедура заканчивает работу.

Далее приводится описание функции ОБЩ_РЕШ(U, $\vec{a}$, $x_0^\downarrow$, c), которая используется при выполнении каждого из указанных шагов, и общего шага процедуры.

Функция ОБЩ_РЕШ(U, $\vec{a}$, $x_0^\downarrow$, c) предназначена для решения следующей задачи.

Пусть задана СЛУ над кольцом R

$$\grave{A}\tilde{o}^\downarrow = b^\downarrow , \tag{9}$$

где $A \in R_{n,n}$, $b^\downarrow \in R^{(n)}$ и пусть ($x_0^\downarrow$, U) – общее решение СЛУ (9), $x_0^\downarrow \in R^{(n)}$, $U \in R_{n,n}$. Необходимо проверить совместность и в случае совместности найти общее решение СЛУ

$$\begin{cases} Ax^\downarrow = b^\downarrow, \\ \vec{a}x^\downarrow = c, \end{cases} \tag{10}$$

где $\vec{a} \in R^{(n)}$, $c \in R$.

Нетрудно видеть, что для любых $\vec{a} \in R^{(n)}$, $c \in R$ СЛУ (10) совместна тогда и только тогда, когда совместно линейное уравнение (ЛУ)

$$(\vec{a}U)y^\downarrow = c - \vec{a}x_0^\downarrow , \tag{11}$$

при этом, если ($y_0^\downarrow$, V) – общее решение ЛУ (11), то общее решение СЛУ (10) имеет вид

$$\left(x_0^\downarrow + Uy_0^\downarrow, U \cdot V\right).$$

Приведем формальное описание функции ОБЩ_РЕШ(U, $\vec{a}$, $x_0^\downarrow$, c).

Входные данные: $U \in R_{n,n}$, $\vec{a} \in R^{(n)}$, $x_0^{\downarrow} \in R^{(n)}$, $c \in R$.

Выходные данные: упорядоченный набор элементов (*flag*, Q_1, Q_2), где *flag* – логическая переменная, $Q_1 \in R^{(n)}$, $Q_2 \in R_{n,n}$. Выходные данные принимают такие значения:

- *flag = true*, $Q_1 = x_0^{\downarrow} + Uy_0^{\downarrow}$, $Q_2 = U \cdot V$ – в случае совместности СЛУ (10),
- *flag = false*, $Q_1 = x_0^{\downarrow}$, $Q_2 = U$ – в противном случае.

1. Начало функции. Вычислить $\vec{g} = \vec{a}U$, $h = c - \vec{a}x_0^{\downarrow}$.

2. Представить вектор $\vec{g}$ = (g_1, …, g_l) в виде: $\vec{g}$ = ($p^{\alpha_1} \cdot g_1'$, …, $p^{\alpha_l} \cdot g_l'$), свободный член h в виде $h = p^{\alpha_h} \cdot h'$, где $g_1', \ldots, g_l', h' \in R^*$ – обратимые элементы кольца R. Положить $\alpha_{min} = min(\alpha_1, \ldots, \alpha_l)$.

3. Проверить выполнение условия

$$\alpha_{min} \le \alpha_h. \tag{12}$$

В случае невыполнения условия (12) положить *flag = false*, $Q_1 = x_0^{\downarrow}$, $Q_2 = U$ и осуществить возврат в точку вызова. В противном случае – перейти к шагу 4.

4. Вычислить

$$y_t = \left(g_t'\right)^{-1} p^{\alpha_h - \alpha_{\min}} h' , \tag{13}$$

где t – порядковый номер координаты вектора $\left(p^{\alpha_1}, \ldots, p^{\alpha_l}\right)$, показатель степени которой равен α_{min}.

5. Положить

$$y_0^{\downarrow} = (0, \ldots, 0, y_t, 0, \ldots, 0)^{\mathrm{T}}. \tag{14}$$

6. Для всех $q \in \overline{1,l} \setminus \{t\}$, вычислить

$$y_t^{(q)} = -\left(g_t'\right)^{-1} p^{\alpha_q - \alpha_{\min}} g_q' ,$$

где индекс t определяется аналогично (13).

7. Положить в качестве общего решения однородного ЛУ (11) матрицу

$$V = \begin{pmatrix} 1 & 0 & \ldots & 0 & y_t^{(1)} & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 & y_t^{(2)} & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \cdots & 0 & y_t^{(l)} & 0 & \ldots & 1 \end{pmatrix}^{\mathrm{T}} . \tag{15}$$

8. Положить *flag = true*, $Q_1 = x_0^{\downarrow} + Uy_0^{\downarrow}$, $Q_2 = U \cdot V$. Возврат в точку вызова.

Отметим, что на основании результатов, изложенных в [19, стр. 133], определенные с помощью формул (14) и (15) значения вектора $y_0^{\downarrow}$ и матрицы V действительно являются частным и общим решением ЛУ (11).

Описание общего шага процедуры.

На первом (z = 1) шаге процедуры положить m = 0, выбрать вектор $\overrightarrow{c_{0,1}}$, вызвать для всех $l \in \overline{0,d-1}$, $\tilde{O} \in \overline{\Delta}_l^1$ функцию ОБЩ_РЕШ($E_{|\overline{X}|,|\overline{X}|}$, $\left(\overrightarrow{c_{0,1}}\right)_{\tilde{O}}$, $0^{\downarrow}$, $p^{d-(l+1)}\left(\overrightarrow{c_{0,1}}\right)_0$), где $E_{|\overline{X}|,|\overline{X}|}$ – единичная матрица над R размера $|\overline{X}| \times |\overline{X}|$, $0^{\downarrow}$ – нулевой вектор длины $|\overline{X}|$.

Пусть теперь $z > 1$ и на предыдущем, (z – 1)-м, шаге процедуры получена матрица C, обладающая свойствами, изложенными в разделе I, и для всех $l \in \overline{0,d-1}$, $\tilde{O} \in \overline{\Delta}_l^1$ определены общие решения $x_0^{\downarrow}\left(\tilde{O},l\right)$ и $U\left(\tilde{O},l\right)$ соответствующих СЛУ вида (3).

1. Если $m = r$, то положить $C_\Sigma = C$, завершить работу процедуры.

Если $m < r$, то с учетом указанного выше порядка следования строк матрицы C выбрать вектор $\overrightarrow{c_{i,j}} = \left(p^i, \overrightarrow{c_{i,j}'}\right)$, где $i \in \overline{0,d-1}$, $j \in \overline{1,r_i}$, удовлетворяющий условию (*).

2. Для всех $l \in \overline{0,d-1}$, $\tilde{O} \in \overline{\Delta}_l^1$ вызвать функцию ОБЩ_РЕШ($U(\tilde{O},l)$, $\left(\overrightarrow{c_{i,j}}\right)_{\vec{O}}$, $x_0^{\downarrow}(\tilde{O},l)$, $p^{d-(l+1)}c_0^{\downarrow}$).

3. Если в результате выполнения предыдущего пункта получено заключение о несовместности хотя бы одной СЛУ (*flag = false* для некоторого $\tilde{O} \in \overline{\Delta}_l^1$, $l \in \overline{0,d-1}$), перейти к следующему, (z + 1)-му, шагу процедуры.

Если в результате выполнения предыдущего пункта получено заключение о совместности всех СЛУ (*flag = true* для всех $l \in \overline{0,d-1}$, $\tilde{O} \in \overline{\Delta}_l^1$), положить $m = m + 1$,

$$\tilde{N} = \begin{pmatrix} C \\ \overrightarrow{c_{i,j}} \end{pmatrix},$$

и перейти к следующему, (z + 1)-му, шагу процедуры. Конец z-го шага.

В результате выполнения изложенной процедуры будет построена матрица C_Σ или сделан вывод о несуществовании матрицы G вида (1) для заданных входных данных. Временная и емкостная сложности этой процедуры приведены в следующем разделе.

Этап построения результирующей матрицы G.

Входными данными для вычислительной процедуры, выполняемой на этом этапе, является матрица C_Σ.

В случае негативного результата на предыдущем этапе следует сделать заключение о том, что для заданной совокупности множеств не существует реализующего ее ПМРС. В противном случае, как указывалось выше, в качестве строк матрицы G выбираются элементы системы образующих модуля решений СЛУ $C_\Sigma x^{\downarrow} = 0^{\downarrow}$ вида (1) над кольцом R.

Временная сложность решения указанной СЛУ составляет $10/3n^3 - 2n^2 - 1/3n$ ЭО (например, используя алгоритм приведения матрицы СЛУ к каноническому виду [19]).

IV Оценки временной и емкостной сложностей основного этапа предложенного алгоритма

Оценим *временную сложность* T – число ЭО, которое выполняется в наихудшем случае при построении матрицы C_Σ на втором этапе алгоритма. Убедимся в справедливости равенства

$$T = \left(6n^2 + 5n - 1\right)\sum_{i=0}^{d-1}\left|\overline{\Delta}_i^1\right|\left(p^d - 1\right)^{\left(\sum_{i=0}^{d-1}\sum_{j=1}^{r_i}\left|A_{i,j}\right| - n\right)} d^n. \qquad (16)$$

Для доказательства соотношения (16) рассмотрим табл. 1, в которой приведены значения временных сложностей соответствующих шагов описанной выше функции ОБЩ_РЕШ(U, $\vec{a}$, $x_0^{\downarrow}$, c).

Таблица 1 – Временные сложности шагов функции ОБЩ_РЕШ(U, $\vec{a}$, $x_0^{\downarrow}$, c)

№	Обращение	Умножение	Вычитание	Сложение
1	-	$n^2 + n$	1	$n^2 - 1$
4	1	2	-	-
6	$n - 1$	$2n - 2$	$n - 1$	-
8	-	$2n^2$	-	$2n^2$
Всего:	n	$3n^2 + 3n$	n	$3n^2 - 1$

Сумма значений, приведенных в последней строке табл. 1, определяет трудоемкость проверки совместности СЛУ (3) для одной матрицы $\tilde{O} \in \overline{\Delta}_l^1$, где $l \in \overline{0,d-1}$, и составляет $6n^2 + 5n - 1$ ЭО. Всего при однократном выполнении общего шага процедуры построения матрицы C_Σ проверяется совместность $\sum_{i=0}^{d-1}\left|\overline{\Delta}_i^1\right|$ таких матриц. Отметим теперь, что в наихудшем случае необходимо перебрать все значения ненулевых элементов матрицы C, что с учетом особенностей ее задания составляет $\left(p^d - 1\right)^{\left(\sum_{i=0}^{d-1}\sum_{j=1}^{r_i}\left|A_{i,j}\right| - n\right)} d^n$ значений. Перемножая три полученных выражения, получаем формулу (16), что и требовалось доказать.

Оценим *емкостную сложность* K – число регистров памяти РАМ, задействованных при построении матрицы C_Σ на втором этапе алгоритма. Отметим, что при выполнении вычислительной процедуры этого этапа необходимо хранить $\sum_{i=0}^{d-1}\left|\overline{\Delta}_i^1\right|$ упорядоченных пар ($x_0^{\downarrow}$, U), $x_0^{\downarrow} \in R^n$, $U \in R_{n,n}$, а также саму матрицу C размера $n \times r$. Итак, выполняется равенство

$$\hat{E} = \left(n^2 + n\right)\sum_{i=0}^{d-1}\left|\overline{\Delta}_i^1\right| + nr .$$

Нетрудно показать, что временная сложность $T_{\text{ПЕР}}$ переборного алгоритма, описание которого приведено в разделе I, составляет

$$\grave{O}_{\text{ÏÅÐ}} = \left(p^d\right)^{nk}\sum_{i=0}^{d}\left|\Sigma_i^0\right|T_0 .$$

V Пример практического применения алгоритма

Рассмотрим пример, который демонстрирует эффективность предложенного алгоритма по сравнению с тривиальным (переборным) алгоритмом.

Пусть необходимо построить ПМРС для совокупности множеств $\Delta^0 = \left\{\Delta_0^0 = \left\{\left\{1,2,3\right\}\right\}; \Delta_1^0 = \left\{\left\{1,2\right\}\right\}\right\}$, где $P = \{1, 2, 3\}$. При этом $P = \{1, \ldots, 4\}$, $p = 2$, $d = 2$.

Нетрудно убедиться в том, что применение вычислительных процедур, изложенных в разделе II, позволяет построить по совокупности множеств Δ^0 совокупность множеств $\Sigma = \{\Sigma_0 = \{\{1,2,3\}\}, \Sigma_1 = \{\{1,2\}\}, \Sigma_2 = 2^P\backslash\{\Sigma_0 \cup \Sigma_1\}\}$. При этом $\Sigma_0^0 = \Sigma_0 = \left\{\left\{1,2,3\right\}\right\}$, $\Sigma_1^0 = \Sigma_1 = \left\{\left\{1,2\right\}\right\}$, $\overline{\Delta}_0^1 = \left\{\left\{1,2\right\},\left\{1,3\right\},\left\{2,3\right\}\right\}$, $\overline{\Delta}_1^1 = \left\{\left\{1,3\right\},\left\{2,3\right\}\right\}$. Матрица C_Σ над кольцом $Z/4$, полученная в результате применения предложенного алгоритма, имеет вид

$$\mathrm{C}_\Sigma = \begin{pmatrix} 1 & 3 & 1 & 2 \\ 2 & 2 & 2 & 0 \end{pmatrix},$$

а матрица G, определяющая искомый ПМРС, равна

$$G = \left(\begin{array}{c|ccc} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 2 \end{array}\right).$$

Согласно равенству (16), для вычисления матрицы C_Σ необходимо 24480 ЕО. Для вычисления матрицы G, как указано в разделе I, необходимо 251 ЕО. Таким образом, общая временная сложность построения ПМРС составляет 24480 + 251 = 24731 ЕО. Временная сложность $T_{\text{ПЕР}}$ вычислительной процедуры переборного алгоритма для тех же исходных данных составляет $3.4 \cdot 10^{10}$ ЕО. Таким образом, выигрыш во временной сложности при применении предложенного алгоритма для приведенного примера равен $T_{\text{ÏÅÐ}} / T \approx 1.41 \cdot 10^6$.

Приведенный пример демонстрирует существенно более высокую вычислительную эффективность предложенного алгоритма по сравнению с тривиальным алгоритмом.

VI Заключение

В статье предложен алгоритм проверки существования и формирования (при условии существования) матриц над кольцом вычетов по примарному модулю, используемых для построения линейных протоколов множественного разделения секрета, реализующих заданную иерархию доступа. Алгоритм разработан на основе аналитического описания конструкций указанных ПМРС, полученного в [17], и позволяет на практике осуществлять проверку существования для заданной совокупности множеств Σ матрицы G вида (1), определяющей протокол множественного разделения секрета, для которого совокупность Σ является иерархией доступа, а также (в случае существования такой матрицы) строить ее в явном виде.

Предложенный алгоритм обобщает известный ранее алгоритм формирования матриц над конечным

полем для построения линейных совершенных протоколов разделения одного секрета [12] и, как показано в статье, имеет меньшую временную сложность по сравнению с тривиальным алгоритмом.

Литература: ***1.*** *Введение в криптографию / Под общ. ред. В. В. Ященко. – М.: МЦНМО: "ЧеРо", 1999. – 272 с.* ***2.*** *Seberry J., Charnes C., Pieprzyk J., Safavi-Naini R. 41 Crypto topics and applications II. Handbook on Algorithms and Theory of Computation, 1998. – P. 1 – 22.* ***3.*** *McLean J. Reasoning about security models // Proceeding IEEE Symposium on privacy and security. – IEEE Computer Society Press. – 1987. – P. 123-131.* ***4.*** *Blakley G. R. Safeguarding cryptographic keys // Proc. AFIPS 1979 National Computer Conference. – N-Y.:1979. – V. 48. – P. 313 – 317.* ***5.*** *Shamir A. How to share a secret // Comm. ACM. – 1979. – V. 22. – № 1. – P. 612 – 613.* ***6.*** *Bertilsson M. Linear codes and secret sharing. – PhD Thesis. – Linkoping University. – 1993.* ***7.*** *Brickell E. F. Some ideal secret sharing schemes // J. Combin. Math. and Combin. Comput. – 1989. – № 9. – P. 105 – 113.* ***8.*** *Blakley G. R., Kabatianski G. A. Linear algebra approach to secret sharing schemes // Preproc. of Workshop on Information Protection.: Moscow, 1993.* ***9.*** *Massey J. L. Minimal codewords and secret sharing // Proc. 6th Joint Swedish-Russian Int. Workshop on Information Protection. – 1993. – P. 276 – 279.* ***10.*** *Ashikhmin A., Barg A. Minimal vectors in linear codes // IEEE Trans. on Inform. Theory. – 1998. – V. 5. – P. 2010 – 2018.* ***11.*** *Ashikhmin A., Barg A. Minimal vectors in linear codes and sharing of secrets // Univ. Bielefeld, SFB 343 Diskrete Strukturen in der Mathematik. – 1994. – Preprint 94 – 113, available from ftp.uni-bielefeld.de.* ***12.*** *van Dijk M. A Linear construction of perfect secret sharing schemes // Advances in Cryptology – EUROCRYPT'94. – Lecture Notes in Comput. Science. – V. 950. – P. 23 – 34.* ***13.*** *Simmons G. J. How to (really) share a secret // Advances in Cryptology – CRYPTO'88, Lecture Notes in Computer Science. – 1989. – Vol. 403. – P. 390 – 448.* ***14.*** *Blundo C., de Santis A., di Crescenzo D., Gaggia A. G., Vaccaro U. Multi-secret sharing schemes // Advances in Cryptology – CRYPTO'94, Lecture Notes in Computer Science. – 1994. – Vol. 839. – P. 150 – 163.* ***15.*** *Алексейчук А. Н., Волошин А. Л. Совершенная схема множественного разделения секрета над кольцом вычетов по модулю m // Реєстрація, зберігання і обробка даних. – 2005. – Т. 7. – № 4. – С. 44 – 53.* ***16.*** *Алексейчук А. Н., Волошин А. Л., Скрипник Л. В. Совершенная схема множественного разделения секрета на основе линейных преобразований над конечным цепным коммутативным кольцом // Материалы международной научной конференции по проблемам безопасности и противодействия терроризму. Интеллектуальный Центр МГУ. 2 – 3 ноября 2005 г. – М.: МЦНМО, 2006. – С. 149 – 154.* ***17.*** *Алексейчук А. Н., Волошин А. Л. Аналитическое описание конструкций протоколов множественного разделения секрета с многоадресным сообщением, реализующих заданную иерархию доступа // Прикладная радиоэлектроника. – 2007. – Т.6. – №3. – С. 391 – 396.* ***18.*** *Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. – Пер. с англ. – М.: Мир, 1979. – 536 с.* ***19.*** *Глухов М. М., Елизаров В. П., Нечаєв А. А. Алгебра. Учебник. В 2-х т. Т. 1. – М.: Гелиос АРВ, 2003. – 336 с.*

УДК 621.391:519.2

ОЦІНКИ ЙМОВІРНОСТЕЙ УЗАГАЛЬНЕНИХ ЛІНІЙНИХ АПРОКСИМАЦІЙ РАУНДОВОЇ ФУНКЦІЇ ГОСТ-ПОДІБНОГО БЛОКОВОГО ШИФРУ

Артур Шевцов

Інститут спеціального зв'язку та захисту інформації НТУУ "КПІ"

Анотація: **Отримані аналітичні верхні межі ймовірностей узагальнених лінійних апроксимацій раундової функції ГОСТ-подібного блокового шифру, які залежать від певних числових параметрів його вузлів заміни. Отримані результати складають основу подальших досліджень в галузі аналізу та обґрунтування стійкості ГОСТ-подібних блокових шифрів відносно методу узагальненого лінійного криптоаналізу.**

Summary: **Analytical upper bounds of generalized linear approximations probabilities of the round function of a GOST-like block cipher are obtained. These bounds depends on some numerical parameters of S-boxes of the given block cipher. Obtained results form the basis for next research in area of analysis and security proving of GOST-like block ciphers against generalized linear cryptanalysis techniques.**

Ключові слова: **ГОСТ-подібний блоковий шифр, узагальнений лінійний криптоаналіз, узагальнена лінійна апроксимація раундової функції.**