УДК 004.056.55

АНАЛИЗ ШИФРА «КУЗНЕЧИК» МЕТОДОМ СВЯЗАННЫХ КЛЮЧЕЙ

Ищукова Е.А., Красовский А.В., Половко И.Ю.

Южный федеральный университет, Институт компьютерных технологий и информационной безопасности, Таганрог, e-mail:an.krasowsckij@gmail.com

Объектом исследования является блочный симметричный шифр «Кузнечик», который является национальным стандартом РФ в области шифрования. Средством исследования является метод связанных ключей, который позволяет оценить надёжность и криптостойкость данного шифра. Результатом работы являются обнаруженные способы применения метода связанных ключей к шифру «Кузнечик», систематизированные дифференциальные свойства блоков шифра, созданные алгоритмы определения значений подключей, а также логические заключения о структурировании и классифицировании подходов к анализу шифра «Кузнечик» с помощью метода связанных ключей. Для проведения тестирования и определения временной сложности был реализован шифр на языках программирования C и на Python. Результаты показали, что к алгоритму «Кузнечик» применим метод анализа связанных ключей, при условии что в используемые раундовые подключи внесено нарушение по определенному принципу. Также показано, что при использовании классической реализации, определенной в стандарте ГОСТ Р 34.12 – 2015, шифр «Кузнечик» остается устойчивым к атакам на основе связанных ключей. Проведенные теоретические расчеты подтверждаются экспериментальными данными, полученными с использованием разработанных и реализованных алгоритмов анализа на основе метода связанный ключей.

Ключевые слова: анализ, связанные ключи, дифференцирование, «Кузнечик», ГОСТ Р34.12-2015

ANALYSIS OF THE CIPHER KUZNECHIK BY THE RELATED KEYS METHOD

Ischukova E.A., Krasovskiy A.V., Polovko I.Yu.

Southern Federal University, Institute of Computer Technologies and Information Security, Taganrog, e-mail: an.krasowsckij@gmail.com

The subject of research is a block symmetric cipher Kuznechik, which is the national standard of the Russian Federation in encryption. The means of investigation is the method of related keys, which makes it possible to evaluate the reliability and cryptographic strength of this cipher. The results of the work are the discovered ways to apply the method of related keys to the Kuznechik, the differential cryptanalysis's properties of the cipher blocks, the algorithms for determining the values of the subkeys, as well as logical conclusions about structuring and classification approaches to the analysis of the cipher Kuznechik using the related key method. To conduct testing and determine the time complexity, a cipher in C and Python was implemented. The results showed that the Kuznyechik algorithm is applicable to the related keys method, provided that the violation is applied to the used round plug-ins according to a certain principle. It is also shown that when using the classical implementation defined in the standard GOST R 34.12 – 2015, the Kuznyechik cipher remains resistant to attacks based on related keys. The carried out theoretical calculations are confirmed by experimental data obtained using developed and implemented analysis algorithms based on the related key method.

Keywords: analysis, related keys, differentiation, Kuznyechik, GOST R34.12-2015

Анализ криптографических алгоритмов представляет собой актуальную и востребованную научную работу. Существует множество способов анализа шифров, и одним из них является метод связанных ключей [1]. Метод связанных ключей (МСК) оценивается как эффективный криптографический подход к анализу. В настоящий момент МСК уже успешно опробован на мировых и национальных стандартах шифрования [2–4], но остаётся только теоретической атакой с высоким уровнем допущений относительно знаний аналитика.

Разработчики современных шифров создают сложные процедуры генерации подключей и учитывают МСК. Также они стремятся разрабатывать шифры, которые будут оптимально реализованы на конкретных вычислительных устройствах. Такой двойственный учёт теоретической и практической стороны оправдан и предоставляет высокую степень криптостойкости, но не разрешает все потенциальные проблемы.

По мнению авторов данной статьи, одной из таких проблем может быть использование основ МСК во взаимосвязи с иными видами атак и анализа. Проблема актуальна и для шифра «Кузнечик». Данный алгоритм является современным национальным стандартом РФ в области шифрования. Определение криптостойкости шифра «Кузнечик» является важным процессом, так как стойкость – это основополагающий фактор при выборе алгоритмов шифрования в рамках комплекса мер обеспечения информационной безопасности.

В связи с положениями, описанными выше, было проведено исследование «Кузнечика» с целью определения стойкости шифра относительно МСК с выделением и изучением уникальных свойств и осо-

бенностей алгоритма (вырабатывая другие подходы к анализу). В исследовании уровень знания аналитика повышался по необходимости.

Достижением данной работы является выявление параметров восстановления ключа для большего количества раундов (максимально 5 и 6) по сравнению с другими работами в данной тематике [5, 6].

В работе определяется и описывается структура шифра, определяются все необходимые для достижения цели свойства шифра. Далее определяются алгоритмы восстановления промежуточных значений процессов шифрования/дешифрования и описывается алгоритм восстановления ключа.

Структура шифра «Кузнечик»

ГОСТ 34.12-2015 [7] «Кузнечик» является блочным симметричным шифром. Он реализован в соответствии с SP сетью, где процесс дешифрования обратен шифрованию. Вход/выход и промежуточные значения имеют размерность 128 бит, ключ имеет размерность 256 бит. Всего «Кузнечик» состоит из 10 блоков: S и $\acute{S}$, p и $\acute{p}$, L и $\acute{L}$, R и $\acute{R}$, X и l. Где блок X представляет из себя блок побитового сложения двух входных значений по модулю два.

Блоки замены байта p и $\acute{p}$ применяются в рамках блоков S и $\acute{S}$ соответственно и имеют размерность 8 бит. «Кузнечик» имеет предустановленные таблицы замены, которые можно считать массивами с индексами. Выходом блоков p и $\acute{p}$ является значение, взятое в таблице замены по индексу равному значению входа.

Блоки замены S и $\acute{S}$ имеют входную/выходную размерность 128 бит. Входное битовое слово разделяется по 8 бит последовательно непрерывно. Каждый полученный байт заменяется с помощью p и $\acute{p}$ блока. Значения после p и $\acute{p}$ блоков организуют выходное слово в соответствии с входным порядком битов. Обозначим логическое разбиение 128-битного текста на 16 байт как $a = a_{15} \| a_{14} \| \ldots \| a_0$, $a \in V_{128}$; Схематичное представление блоков S и $\acute{S}$ приведено на рис. 1.

Блок l имеет размерность входа/выхода 128/8 бит соответственно. Выход генерируется в соответствии с уравнением $l(a)$. Умножение и сложение происходят в поле $GF(2)[x]/p(x)$, где $p(x) = x^8 + x^7 + x^6 + x + 1$.

$$l(a) = 145 * a_{15} + 32 * a_{14} + 133 * a_{13} + 16 * a_{12} + 194 * a_{11} +$$

$$+192 * a_{10} + 1 * a_9 + 251 * a_8 + 1 * a_7 + 192 * a_6 + 194 * a_5 +$$

$$+16 * a_4 + 133 * a_3 + 32 * a_2 + 148 * a_1 + 1 * a_0 .$$

Блоки R и $\acute{R}$ имеют размерность входа/выхода 128 бит. Они используют блок l для вычисления нового старшего байта.

$$R(a_{\text{вход}}) = l\left(a^{15}_{\text{вход}} \| a^{14}_{\text{вход}} \| \ldots \| a^{0}_{\text{вход}}\right) \| a^{15}_{\text{вход}} \| a^{14}_{\text{вход}} \| \ldots \| a^{1}_{\text{вход}}$$

$$\acute{R}(a_{\text{вход}}) = a^{14}_{\text{вход}} \| a^{13}_{\text{вход}} \| \ldots \| l\left(a^{14}_{\text{вход}} \| a^{13}_{\text{вход}} \| \ldots \| a^{1}_{\text{вход}} \| a^{15}_{\text{вход}}\right)$$

Блоки L и $\acute{L}$ имеют размерность входа/выхода 128 бит. Они используют R и $\acute{R}$ соответственно последовательно 16 раз.

$$L(a_{\text{вход}}) = R^{16}(a_{\text{вход}})$$

$$\acute{L}(a_{\text{вход}}) = \acute{R}^{16}(a_{\text{вход}})$$

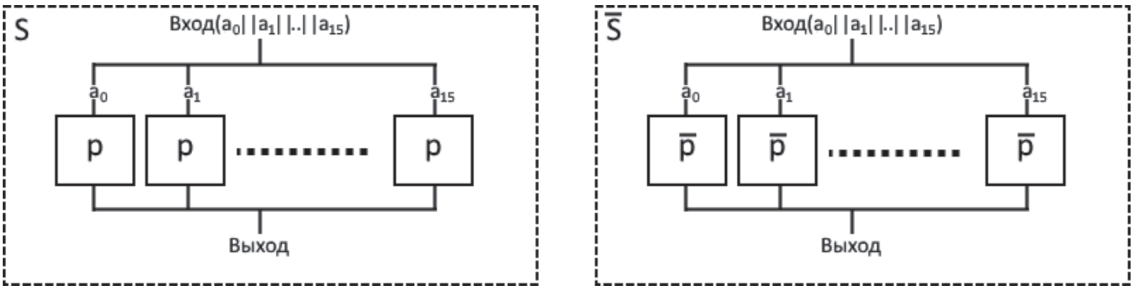

Рис. 1. Схематичное представление блоков S и Ś, где показан процесс преобразования входного значения $a = a_{15} \| a_{14} \| \ldots \| a_0$

Процессы шифрования и дешифрования отображаются формулами (1, 2) соответственно. Обозначим $C \in V_{128}$ как закрытый текст и $P \in V_{128}$ как открытый текст.

$$C = X(K_{10})LSX(K_9)\ldots LSX(K_1)(P), \quad (1)$$

$$P = X(K_1)S'L'X(K_2)\ldots S'L'X(K_{10})(C). \quad (2)$$

Ключ шифра имеет размерность 256 бит, а подключ 128 бит. Генерируется 10 подключей для 9 раундов и заключающего X блока. Для выработки подключей используется 32 постоянных продекларированных в стандарте значений C_i. Обозначим старшие 128 бит ключа шифрования как K_1, а младшие K_2.

$$C_i := L(i), i := 1,2,\ldots,32,$$

$$F[C_n](K_{i-1}, K_i) = \left(LSX(C_n)(K_{i-1}) \oplus K_i, K_{i-1}\right),$$

$$\left(K_{2i+1}, K_{2i+2}\right) = F\left[C_{8(i-1)+8}\right]\ldots F\left[C_{8(i-1)+1}\right]\left(K_{2i-1}, K_{2i}\right), i := 1,\ldots,4.$$

Свойства шифра «Кузнечик»

При исследовании структуры шифра на наличие возможных способов применения МСК было установлено, что необходимо анализировать промежуточные дифференциалы процессов шифрования/дешифрования. Промежуточное значение (ПЗ) – это 128-битное слово, которое является результатом вычисления предыдущего блока и входом последующего.

В данном разделе рассматриваются необходимые для МСК дифференциальные свойства (ДС). Дифференциальными свойствами обладает как шифр в общем, так и в частности блоки и их объединения. Шифр имеет 5 основных общих блоков S и $\acute{S}$, L и $\acute{L}$, X и, следовательно, далее рассматривается их ДС.

Блок X представляет из себя простой и понятный элемент шифра. Он обладает известной структурой, и его свойства соответствуют ДС побитового сложения по модулю два.

Блоки L и $\acute{L}$, при рассмотрении с позиции ДС, обладают свойством дистрибутивности. Обозначим значения $a, b \in V_{128}$ как входные значения двух процессов преобразования с помощью L и $\acute{L}$. Схематично данное ДС изображено на рис. 2.

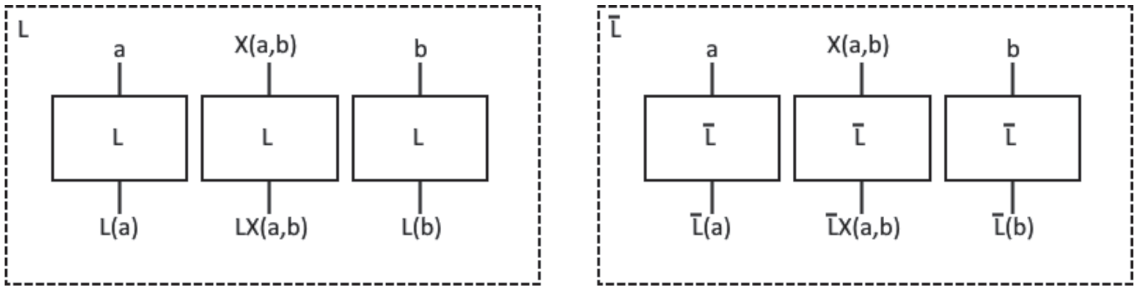

Рис. 2. Схематичное представление ДС блоков L и Ĺ, где X(a, b) является дифференциалом входных значений

$$L(a \oplus b) = L(a) \oplus L(b)$$

$$L'(a \oplus b) = L'(a) \oplus L'(b)$$

Блоки S и $\acute{S}$ обладают ДС блоков p и $\acute{p}$. При изучении ДС блоков p и $\acute{p}$ было выявлено, что разные дифференциалы входов образуют неодинаковое количество неповторимых выходных дифференциалов. Данное ДС для уникальности обозначим как свойство неравномерности распределения (СНР). Далее рассматривается СНР p блока.

a/b	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.	1	108	102	109	109	107	100	107	107	106	105	110	109	105	111	104
1.	110	110	105	108	106	107	106	102	110	110	107	104	113	105	109	109
2.	102	102	110	106	108	105	98	105	111	107	105	108	109	102	104	106
3.	112	113	109	100	114	113	110	103	111	107	108	107	108	107	107	108
4.	105	107	107	106	106	109	110	104	108	103	106	111	104	109	111	105
5.	106	110	108	108	104	107	103	106	114	110	104	108	104	112	104	109
6.	110	107	111	103	105	104	108	114	109	110	103	106	106	107	107	108
7.	110	106	105	101	108	104	107	108	104	112	109	107	107	104	107	106
8.	103	110	104	108	114	102	103	107	106	107	105	107	110	100	110	102
9.	104	108	105	108	110	105	107	113	101	107	109	113	107	105	100	106
A.	113	105	109	111	109	101	102	107	110	100	106	110	107	103	105	106
B.	110	103	105	106	104	101	110	110	110	112	110	113	105	111	108	105
C.	111	106	111	105	107	109	99	104	103	105	106	111	109	103	106	108
D.	109	106	113	102	103	112	108	104	105	111	106	106	105	105	111	105
E.	105	112	107	105	104	109	106	109	102	107	109	107	109	106	103	104
F.	106	111	108	112	106	107	114	107	113	112	112	113	103	108	109	112

Рис. 3. СНР p блока с таблицей замены соответствующей стандарту, где минимальные возможные и невозможные повторения выходного дифференциала обозначаются цветом

СНР был практически подсчитан и представлен на рис. 3, где $a,b \in V_4$ – это левая и правая часть входного дифференциала соответственно. На данном рисунке значение, соответствующее входному дифференциалу, обозначает количество уникальных выходных дифференциалов, полученных из входного.

Как видно выше, минимальное значение повторений 98, а максимальное 114. Выходные дифференциалы входного дифференциала равного 0 являются уникальными. Выходные дифференциалы d_i от входного 0x26 образуются в соответствии с формулой

$$d_i = p(i) \oplus p(i \oplus 0x26), i = 0,1,\ldots,255. \quad (3)$$

Другим свойством p и p' блоков является неравномерность распределения повторений (СНП). СНП заключается в неравномерности повторений выходных дифференциалов. СПН следует из СНР. Так, известно, что для любого входного дифференциала существует 256 выходных дифференциалов. Если обратить внимание на рис. 3, то становится очевидно, что какие-то уникальные выходные дифференциалы должны повторяться.

Пусть $d_i^{\text{вх}}$ входной дифференциал p блока, а $\left[d_i^{\text{вх}}\right]$ список выходных дифференциалов полученных от $d_i^{\text{вх}}$ по формуле (3). Обозначим $\left[\left[d_i^{\text{вх}}\right]\right]$ как список уникальных значений в списке $\left[d_i^{\text{вх}}\right]$. В таком случае СНП можно определить как распределение повторений значений из списка $\left[\left[d_i^{\text{вх}}\right]\right]$ в списке $\left[d_i^{\text{вх}}\right]$.

В основном СНП полезен для МСК из-за количества повторений в 2 раза. Это наиболее распространённое повторение, так как здесь используются дифференциалы.

Алгоритм восстановления промежуточных значений

Для восстановления ключа шифрования/дешифрования требуется не дифференциал ПЗ, а его значение. Для получения значения ПЗ из его дифференциала был разработан алгоритм восстановления значений ПЗ (АВЗ).

АВЗ основан на СНП блока p, так как он может быть использован при шифровании и дешифровании и позволяет стандартизировать подход к анализу. СНП заключается в p блоке, а данный блок используется в S блоке. Таким образом, АВЗ позволяет восстанавливать значение на входе и выходе S блока. Для применения АВЗ необходим известный дифференциал ПЗ на входе и выходе, разбитые последовательно и непрерывно на байты. Обозначим данные дифференциалы на входе и выходе как $D^{\text{вх}}$ и $D^{\text{вых}}$ соответственно, а их байт как $d_i^{\text{вх}}$ и $d_i^{\text{вых}}$. Далее следует распределить разбитые байты в соответствии с формулой

$$\left(d_0^{\text{вх}}, d_0^{\text{вых}}\right), \left(d_1^{\text{вх}}, d_1^{\text{вых}}\right), \ldots, \left(d_{15}^{\text{вх}}, d_{15}^{\text{вых}}\right). \quad (4)$$

Все полученные пары необходимо соотнести с СНП p блока и получить потенциальные значения пар входов и выходов. Далее следует собрать все пары на входе/выходе и соотнести их соответственно с входом/выходом. В заключение следует ском-

бинировать все возможные пары значений ПЗ, которые образуют $D^{вх}$ / $D^{вых}$. Выявлено, что минимально возможно восстановить 2^{16} пар значений ПЗ при использовании АВЗ, а максимально 2^{48}.

Алгоритм восстановления ключа

При восстановлении ключей с помощью МСК было установлено, что шифр «Кузнечик» можно представить как криптографический алгоритм с уменьшенным количеством раундов и с нарушенным принципом генерации подключей. Распределим изучаемые представления:

1. Шифр с количеством раундов < 9, выработка подключей нарушено – $К_1$.
2. Шифр с количеством раундов 10, выработка подключей нарушена – $К_2$.
3. Шифр с количеством раундов < 9, стандартная выработка подключей – $К_3$.
4. Шифр с количеством раундов 10, стандартная выработка подключей – $К_4$.

Для всех $К_i$, i = 1, 2, 3, 4 применяется один и тот же алгоритм. Его реализация требует минимум три процесса дешифрования, где подключи взаимосвязаны. Открытый и закрытый текст известны аналитику. Взаимосвязь подключей известна для пар процессов дешифрования, как отображено на рис. 4.

Восстановление ключа совершается в два этапа. На первом этапе в соответствии с рис. 4 используется пара (i, i + 1) процессов дешифрования, а на втором (i, i + 2).

Для первого этапа взаимосвязь подключей должна иметь вид, который позволит восстановить дифференциал ПЗ на входе и выходе последнего S блока. Это предоставит условия для применения АВЗ и восстановления возможных первых подключей.

Для второго этапа взаимосвязь подключей должна иметь вид, который позволит восстановить дифференциал ПЗ на входе и выходе предпоследнего S блока. Учитывая знание первого подключа для i-го процесса дешифрования, можно применить АВЗ и восстановить возможные вторые подключи. После восстановления для i-го процесса дешифрования списка возможных первых и вторых подключей, следует их скомбинировать попарно и проверить.

Изучая все представления шифра «Кузнечик» $К_i$, i = 1, 2, 3, 4, было выявлено, что $К_1$ и $К_2$ идентичны. Для представления $К_3$ были найдены способы применения алгоритма восстановления ключа для пяти и шести подключей с разной сложностью. Представление $К_4$ оказалось абсолютно стойким для МСК в виде, представленном в данной работе. Известно, что $К_4$ соответствует реализации шифра «Кузнечик» по стандарту ГОСТ Р 34.12 – 2015.

Временная сложность и количество восстановленных мастер-ключей разных представлений отображены в таблице. В таблице используются следующие обозначения и сокращения: Мин/Среднее/Макс – это минимальное/среднее/максимальное значение, с – количество восстановленных мастер ключей, символ «–» обозначает неопределённость, t – это количество затрачиваемого времени для вычисления соответствующего количества восстановленных мастер-ключей. Использовался ноутбук HP Pavilion G6, с процессором AMDA10-4600MAPU 2.30 GHz, ОЗУ 8 ГБ и Windows 10x64.

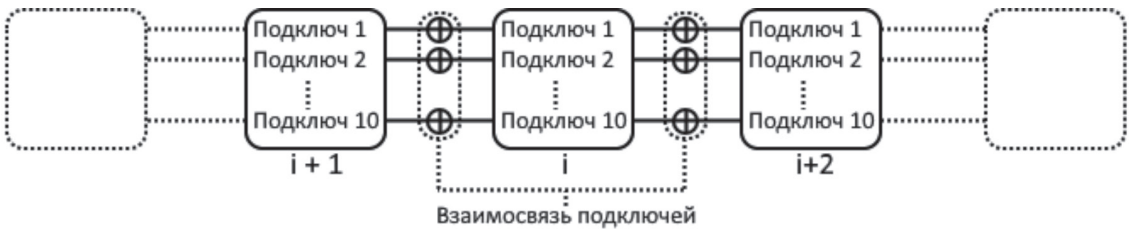

Рис. 4. Схематичное представление взаимосвязи подключей процессов дешифрования, где каждый процесс дешифрования представлен прямоугольным блоком

Вычислительная и временная сложность восстановления ключа

Представление	Мин. с	Макс. с	Среднее с	Мин. t	Среднее t
$К_1$ и $К_2$	2^{16}	2^{96}	2^{24}	16 с.	2 ч. 11 мин.
$К_3$, 5	2^{16}	$103*2^{96}$	2^{25}	16 с.	1 ч. 33 мин.
$К_3$, 6	2^{16}	2^{222}	–	16 с.	–

Результатом работы являются обнаруженные способы применения метода связанных ключей к «Кузнечику», систематизированные дифференциальные свойства блоков шифра, созданные алгоритмы определения значений подключей, а также логические заключения о структурировании и классифицировании подходов к анализу шифра «Кузнечик» с помощью метода связанных ключей.

Список литературы

1. Eli Biham, Orr Dunkelman, Nathan Keller. A Simple Related-Key Attack on the Full SHACAL-1 [Электронный ресурс]. – URL: http://u.math.biu.ac.il/~nkeller/article-842.pdf (дата обращения: 17.04.2018).

2. Vladimir Rudskoy. On zero practical significance of «Key recovery attack on full GOST block cipher with zero time and memory» [Электронный ресурс]. – URL: https://eprint.iacr.org/2010/111.pdf (дата обращения: 17.04.2018).

3. Biryukov A., Khovratovich D. Related-key Cryptanalysis of the Full AES-192 and AES-256 [Электронный ресурс]. – URL: http://eprint.iacr.org/2009/317.pdf (дата обращения: 17.04.2018).

4. Пудовкина М.А., Хоруженко Г.И. Атака на шифросистему ГОСТ 28147-89 с 12 связанными ключами [Электронный ресурс]. – URL: http://kaf42.mephi.ru/wp-content/uploads/2015/12/mvk88.pdf (дата обращения: 17.04.2018).

5. Ищукова Е.А., Красовский А.В., Бабенко Л.К. Оценка стойкости шифра «Кузнечик» с использованием метода связанных ключей // Фундаментальные исследования. – 2016. – № 11–4. – С. 698–703.

6. Ищукова Е.А., Красовский А.В. Возможность применения метода связанных ключей к анализу алгоритма «Кузнечик» // Вестник научных конференций. – 2016. – № 11–6(15). – С. 103–104.

7. ГОСТ 34.12.-2015 «Кузнечик» [Электронный ресурс]. – URL: http://wwwold.tc26.ru/standard/gost/GOST_R_3412-2015.pdf (дата обращения: 17.04.2018).