15. *Mohammad Momani*. Bayesian Fusion Algorithm for Inferring Trust in Wireless Sensor Networks, *Journal of Networks*, 2010, No. 5 (7), pp. 815-822. DOI: 10.4304/jnw.5.7.815–822.
16. *Elmar Schoch, Michael Feiri, Frank Kargl, Michael Weber.* Simulation of Ad Hoc Networks: ns–2 compared to JiST/SWANS // SIMUTools. Marseille, France, 2008.
17. *Shelby Z., Bormann C.* 6LoWPAN: The Wireless Embedded Internet, *Wiley Series on Communications Networking & Distributed Systems*, 2010, pp. 245.
18. *Basan A.S., Basan E.S., Makarevich O.B.* Programma analiza dannykh i vychisleniya doveriya v besprovodnoy sensornoy seti. Svidetel'stvo o gosudarstvennoy registratsii programmy dlya EVM №2016615606, 2016 g. [Program data analysis and computing of trust in wireless sensor networks. The certificate of state registration of computer programs №2016615606, 2016].
19. *Abramson N.* The Throughput of Packet Broadcasting Channels, *IEEE Transactions on Communications*, 1977, Vol. 25, No. 1, pp. 117-128.
20. *Ho J.W.* Zone–based trust management in sensor networks, *in IEEE International Conference on Pervasive Computing and Communications*, 2009, pp. 1-2.
21. *Renjian Feng, Xiaona Han, Qiang Liu, and Ning Yu*. A Credible Bayesian–Based Trust Management Scheme for Wireless Sensor Networks, *Hindawi Publishing Corporation International Journal of Distributed Sensor Networks*, pp. 1-9. DOI: http://dx.doi.org/10.1155/2015/678926.
22. *Chen–xu Liu, Yun Liu, and Zhen–jiang Zhang*. Improved Reliable Trust–Based and Energy–Efficient Data Aggregation for Wireless Sensor Networks, *Hindawi Publishing Corporation International Journal of Distributed Sensor Networks Volume 2013*, pp. 1-11. DOI: http://dx.doi.org/10.1155/2013/652495.
23. *Ganeriwal S., Balzano L.K., and Srivastava M.B.* Reputationbased framework for high integrity sensor networks, *ACM Trans. Sen. Netw*., 2008, Vol. 4, No. 3, pp. 1-37.

Статью рекомендовал к опубликованию д.т.н., профессор Н.И. Витиска.

Басан Елена Сергеевна – Южный федеральный университет; e-mail: ebasan@sfedu.ru; 347922, г. Таганрог, ул. Чехова, 2; тел.: +79515205488; кафедра безопасности информационных технологий; к.т.н.; ассистент**.**

Басан Александр Сергеевич – e-mail: asbasan@sfedu.ru; тел.: +79885370958; кафедра безопасности информационных технологий; к.т.н.; доцент.

Макаревич Олег Борисович – e-mail: mak@tsure.ru; тел.: +78634361518; кафедра безопасности информационных технологий; д.т.н.; профессор.

Basan Elena Sergeevna – Southern Federal University; e-mail: ebasan@sfedu.ru; 2, Chekhov street, Taganrog, 347922, Russia; phone: +79515205488; the department of information security; cand. of eng. sc.; assistant.

Basan Alexander Sergeevich – e-mail: asbasan@sfedu.ru; phone: +79885370958; the department of information security; cand. of eng. sc.; associate professor.

Makarevich Oleg Borisovich – e-mail: mak@tsure.ru; phone: +78634361518; the department of information security; dr. of eng. sc.; professor.

УДК 004.021 DOI 10.23683/2311-3103-2017-5-25-37

Л.К. Бабенко, Е.А. Ищукова, Е.А. Толоманенко

ДИФФЕРЕНЦИАЛЬНЫЙ АНАЛИЗ ШИФРА КУЗНЕЧИК*

Целью данной работы является исследование, разработка и реализация алгоритма Кузнечик, который является частью стандарта ГОСТ Р 34.12-2015, а также алгоритма для его дифференциального анализа. В ходе проведения исследований рассмотрен алгоритм Кузнечик, разработана рабочая программа шифрования и расшифрования на основе данно-

* Работа выполнена при поддержке гранта РФФИ №17-07-00654-а.

го алгоритма. Для проведения анализа трех раундов шифра впервые была предложена схема построения раундовых дифференциалов, основанная на свойствах нелинейного преобразования S и линейного преобразования L. С использованием предложенных разностных характеристик был разработан и реализован алгоритм нахождения правильных пар текстов, подтверждающий работоспособность предложенного способа анализа. В результате выполнения работы разработаны, реализованы и протестированы алгоритм шифрования Кузнечик, а также алгоритм нахождения пар текстов для дифференциального анализа трех раундов шифрования. Программы выполнены на языке программирования C++ в среде разработки Microsoft Visual Studio C++. Результаты исследований подробно описаны, показаны на примерах, структурированы в хронологической последовательности и наглядно отображены в виде иллюстраций, таблиц и схем в тексте данной статьи. Полученный алгоритм шифрования и его программную реализацию можно использовать для зашифрования и расшифрования данных. Разработанный алгоритм нахождения правильных пар текстов для дифференциального анализа и его программную реализацию можно использовать в дальнейшем для продолжения исследований алгоритма Кузнечик и блочных шифров, в целом, а также для усовершенствования метода дифференциального анализа, увеличения количества исследуемых раундов и нахождения оптимизированного метода анализа алгоритма Кузнечик.

Криптография; блочный шифр; SP-сеть; криптоанализ; дифференциальный криптоанализ; шифр "Кузнечик"; ГОСТ Р 34.12-2015.

L.K. Babenko, E.A. Ishchukova, E.A. Tolomanenko

DIFFERENTIAL ANALYSIS OF CIPHER "KUZNYECHIK"

The aim of this work is an investigation, development and implementation of cipher Kuznyechik, which is a part of standard GOST R 34.12-2015, and an algorithm for its differential analysis. Algorithm Kuznyechik has been considered herein, a working program for encryption and decryption based on this algorithm has been developed. For the analysis of the three rounds of cipher, a scheme for constructing round differentials has been firstly proposed based on the properties of nonlinear transformation S and linear transformation L. Using the proposed difference characteristics, an algorithm for finding the correct pairs of texts has been developed and implemented, confirming the operability of the proposed method of analysis. As a result of this work, the encryption algorithm and the algorithm for finding the correct pairs of texts for differential analysis of three rounds of encryption has been developed, implemented and tested. The programs are implemented in the programming language C ++ in the Microsoft Visual Studio C ++ development environment. The research results are described in details, illustrated in examples, structured in chronological order and visually displayed in the form of illustrations, tables and diagrams in the text of this article. The resulting encryption algorithm and its software implementation can be used to encrypt and decrypt data. The developed algorithm for finding the correct pairs of texts for differential cryptanalysis and its software implementation can be used in the future to continue the investigation of algorithm Kuznyechik, and block ciphers, as a whole, and for improving the method of differential cryptanalysis, increasing the number of rounds studied, and finding an optimized method for analyzing the Kuznyechik algorithm.

Cryptography; block cipher; SP-network; cryptanalysis; differential cryptanalysis; cipher Kuznyechik; GOST R 34.12-2015.

Введение. Алгоритм шифрования Кузнечик был выбран в качестве стандарта ГОСТ Р 34.12-2015 и официально вступил в силу 1 января 2016 года, поэтому разработка и реализация программно-ориентированных алгоритмов для его использования и анализа являются *актуальными* [1, 2].

Данная работа выполнялась с целью разработки и реализации алгоритма для анализа шифра Кузнечик. В процессе выполнения исследований был разработан и реализован алгоритм шифрования Кузнечик, изучен метод дифференциального криптоанализа, разработан и реализован метод нахождения правильных пар текстов для проведения дифференциального анализа трех раундов шифра Кузнечик.

На основе выполненной программной реализации алгоритма шифрования Кузнечик получены экспериментальные данные, отражающие результаты работы метода нахождения пар текстов для дифференциального анализа трех раундов шифрования. Все результаты приведены в качестве примеров и описаны в данной статье.

Данная работа посвящена изучению алгоритма Кузнечик, являющегося частью принятого стандарта ГОСТ Р 34.12-2015 с использованием метода дифференциального криптоанализа. Данная задача является актуальной вследствие новизны данного стандарта.

Постановка задачи. Разработать, реализовать и исследовать алгоритм для выполнения дифференциального анализа шифра Кузнечик, сокращенного до 3 раундов.

1. Описание алгоритма шифрования Кузнечик. Алгоритм шифрования Кузнечик представляет собой симметричный блочный шифр с длиной блока равной 128 бит и длиной ключа равной 256 бит.

Шифр Кузнечик, в отличие от своих предшественников, имеет в основе не сеть Фейстеля, а SP-сеть. Использование SP-сети позволяет выполнить преобразования над всем входным блоком целиком, а не только над его половиной.

Процесс шифрования состоит из нескольких раундов, каждый из которых включает в себя несколько преобразований, а именно три: сложение по модулю 2 (xor) с раундовым ключом, замена с помощью блоков подстановок и линейное преобразование.

Раундовые ключи, их десять, вырабатываются на основе 256-битного мастер-ключа. Первые два ключа получаются путем разбиения мастер-ключа пополам, а последующие восемь при помощи восьми раундов сети Фейстеля. В каждом раунде осуществляются: xor ключа с раундовой константой, преобразование с помощью блока подстановок, линейное преобразование. Раундовую константу получаем из применения к номеру раунда линейного преобразования [1–3].

Один раунд шифрования можно представить так, как показано на рис. 1 [4]. Алгоритм шифрования Кузнечик содержит в себе девять аналогичных раундов шифрования. Для каждого раунда из заданного 256-битного мастер-ключа вырабатывается соответствующий раундовый 128-битный ключ с помощью сети Фейстеля. Этот ключ перемешивается путем операции сложения по модулю два с соответствующим блоком данных (рис. 2). Так же для выработки ключа нам понадобится число, называемое раундовой константой *C*, ее получают, применяя преобразование L к номеру раунда – от 1 до 32. На вход сети Фейстеля сначала подаются половинки мастер ключа, а затем выработанные ключи. В качестве левой части подается ключ с индексом 2i, а в качестве правой части ключ с индексом i. Правая часть проходит функцию F (рис. 3) и складывается по модулю 2 с левой частью, затем половинки меняются местами. Подобные преобразования повторяются 8 раз, и мы получаем новую пару ключей. Так вырабатываются ключи с 3-его по 10-ый [1, 2].

Следующим этапом осуществляется преобразование при помощи блока подстановок S. Блок подстановок описан в стандарте. 128-битный блок данных поступает на вход преобразования S, и разбивается на шестнадцать байтов. Каждый байт входного блока – это индекс значения, находящегося в блоке подстановок. Таким образом, на выходе преобразования S будет набор байтов, находящихся по соответствующим индексам в заданном блоке подстановок.

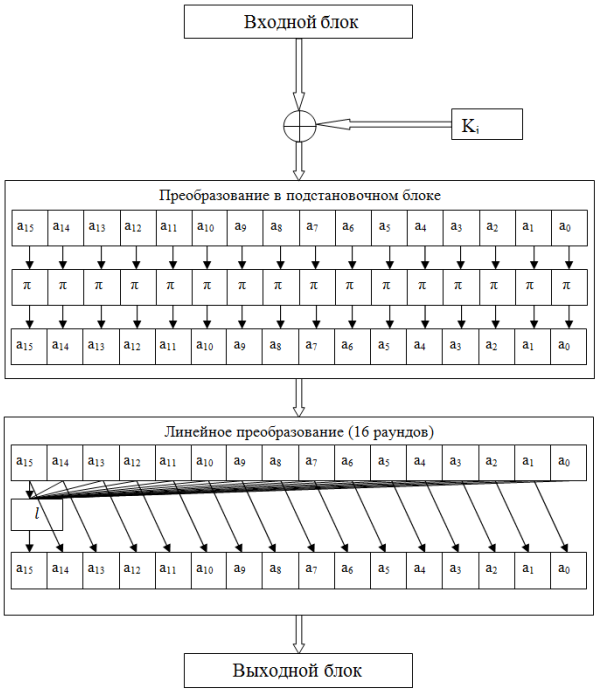

Рис 1. Один раунд шифрования

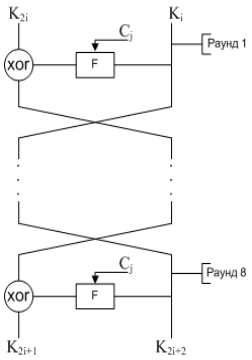

Рис. 2. Сеть Фейстеля для выработки раундовых ключей

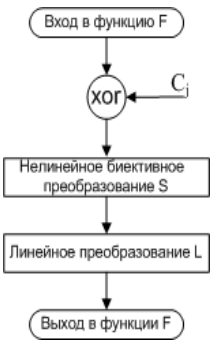

Рис. 3. Функция F, используемая в сети Фейстеля

Далее этот блок подается на вход линейного преобразования L, в котором блок разбивается на 16 байт. Каждый байт умножается на соответствующий ему коэффициент, значение коэффициентов также обозначены в описании стандарта. После операции умножения выполняется операция сложения всех 16-ти элементов между собой. Все расчеты в данном преобразовании производятся в поле Галуа по модулю неприводимого многочлена $x^8 + x^7 + x^6 + x + 1$, коэффициенты также являются элементами поля. Затем полученное число записывается 16-тым байтом, а все остальные байты сдвигаются на один байт вправо, тем самым вытесняя самый крайний правый байт. Подобные расчеты проводятся 16 раз, в результате получаем новое 16-ти байтное значение. Так выполняется один раунд зашифрования. Остальные 8 раундов выполняются абсолютно аналогично. После сложения по модулю два с последним, 10-ым, раундовым ключом мы получаем искомый шифртекст [1–3].

Операция расшифрования выполняется по такому же принципу, но в обратном порядке и с использованием раундовых преобразований, инверсных к тем, что использовались при зашифровании.

2. Разработка и реализация алгоритма для анализа шифра Кузнечик с использованием метода дифференциального криптоанализа. Дифференциальный криптоанализ – один из наиболее распространенных методов криптоанализа симметричных блочных шифров. Метод дифференциального криптоанализа основан на исследовании и изучении преобразований разностей между шифруемыми значениями на различных раундах шифрования. Разность между шифруемыми значениями, как правило, получается вследствие применения операции побитового сложения по модулю два [5–18].

Рассмотрим метод дифференциального криптоанализа шифра Кузнечик. На вход алгоритма подается пара открытых текстов. Обозначим пару открытых текстов как X и X', и тогда их дифференциал обозначим как $\Delta X=X \oplus X'$. Пару выходов, соответствующую открытым текстам обозначим как Y и Y', а их дифференциал обозначим как ΔY. Ключ не может повлиять на дифференциалы, так как при сложении по модулю два все его биты будут взаимно уничтожены – $X \oplus K_i \oplus X' \oplus K_i$. Преобразование L в данном алгоритме является линейным и не может повлиять на результат. Поэтому при использовании метода дифференциального криптоанализа внимание нужно обратить на нелинейное биективное преобразование S – блок подстановки. Дифференциалы, поступающие на вход блока подстановки обозначим как ΔA, а дифференциалы, получаемые на выходе блока подстановки обозначим как ΔC [5, 19].

Для анализа алгоритма нам необходимо составить таблицу для блока подстановок S, строки которой содержат входные значения ΔA в блок подстановок, а столбцы – соответствующие значения ΔC, получаемые на выходе из блока подстановок. На пересечении строк и столбцов отображается количество пар дифференциалов ΔA/ΔC имеющих данные входную и выходную разности. Иными словами, мы получим значение, которое покажет, с какой вероятностью при заданном дифференциале ΔA на выходе из блока подстановок S будет получен конкретный дифференциал ΔC. Часть таблицы вероятностей отображена в табл. 1 [5, 20].

Целью криптоанализа является компрометация раундового ключа, которая в данном случае основана на том факте, что для заданного ΔAне все значения ΔC равновероятны, как видно из табл. 1, вероятность может быть равна 2/256, 4/256, 6/256 и 8/256, а также может быть равной нулю. Комбинация дифференциалов ΔA и ΔC позволяет предположить значения $A \oplus K_i$ и $A' \oplus K_i$. При известных A и A' это позволяет определить K_i [2, 3, 20].

Таблица 1

Часть таблицы вероятностей, составленной для блока подстановок S в процессе дифференциального анализа

ΔC / ΔA	0	1	2	…	3e	3f	…	fe	ff
0	256/ 256	0/256	0/256	…	0/256	0/256	…	0/256	0/256
1	0/256	0/256	2/256	…	2/256	4/256	…	0/256	2/256
2	0/256	4/256	0/256	…	0/256	0/256	…	4/256	2/256
3	0/256	2/256	0/256	…	6/256	2/256	…	0/256	0/256
…	…	…	…	…	…	…	…	…	…
a5	0/256	0/256	0/256	…	0/256	2/256	…	2/256	0/256
a6	0/256	4/256	2/256	…	0/256	0/256	…	0/256	2/256
…	…	…	…	…	…	…	…	…	…
fe	0/256	0/256	2/256	…	2/256	0/256	…	2/256	0/256
ff	0/256	2/256	2/256	…	0/256	4/256	…	0/256	0/256

Рассмотрим дифференциальный анализ трех раундов шифрования с помощью алгоритма шифрования Кузнечик. Схема анализа трех раундов шифрования показана на рис. 4.

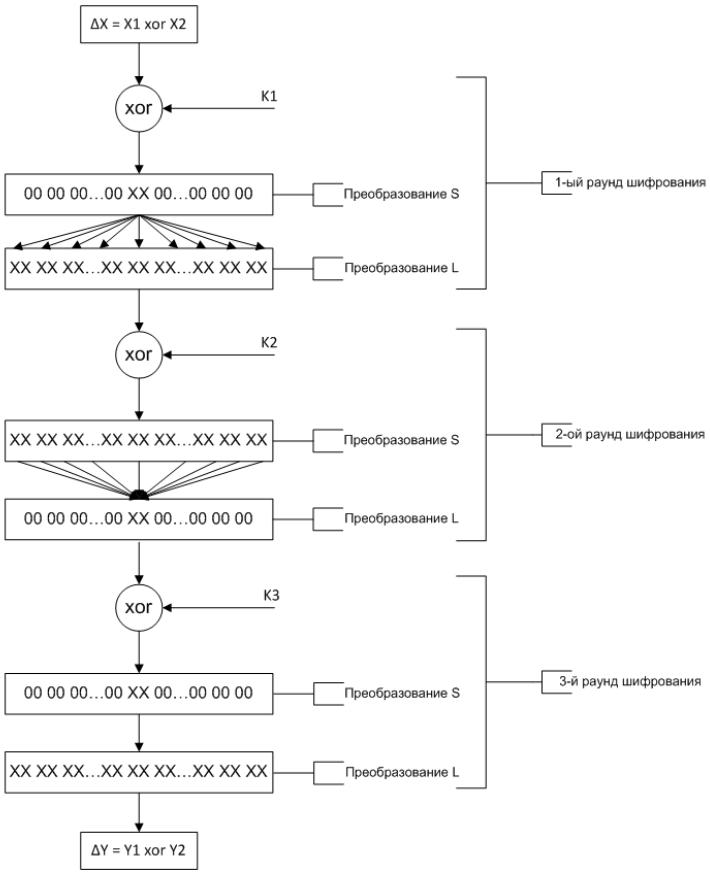

Рис. 4. Схема дифференциального анализа трех раундов шифрования

Первый раунд: генерируется случайный текст X, выбирается дифференциал ΔA, представляющий собой один байт из шестнадцати, текст X', парный X и отличающийся от него лишь значением выбранного дифференциала, находится из следующего выражения: $X' = X \oplus \Delta A$. При $A \oplus K_i \oplus A' \oplus K_i$ ключ уничтожается и данный дифференциал подается на вход нелинейного биективного преобразования S, где изменяется на другой байт в соответствии с таблицей вероятностей, построенной для данного нелинейного преобразования. Результат преобразования S, один полученный байт, подается на вход линейного преобразования L, в результате которого он раскладывается на 16-ти байтное значение (128 бит).

Второй раунд: Полученное в результате преобразования L значение дифференциала складывается со вторым раундовым ключом, аналогично первому раунду ключ уничтожается – $A \oplus K_i \oplus A' \oplus K_i$. На вход нелинейного преобразования S поступает 16-ти байтный блок, каждый байт которого изменится на другой в соответствии с таблицей вероятностей. На вход линейного преобразования L второго раунда поступит данный 16-ти байтный блок с наиболее вероятными значениями, и в результате мы получим однобайтовое значение, 16 байт свернутся в один байт.

Третий раунд: Полученное во втором раунде значение дифференциала складывается с третьим раундовым ключом, который по аналогии с первым и вторым раундом уничтожается. Байтовое значение дифференциала в соответствии с таблицей вероятностей меняется на другое вероятное значение. Затем в результате преобразования L этот байт раскладывается на 16-байтное значение и мы получаем дифференциал на выходе по итогам трех раундов шифрования.

В схеме, рассмотренной на рис. 4, в общей сложности блок замены затрагивается 18 раз. Даже если предположить, что для каждого блока замены будет использована наименьшая вероятность, равная $\frac{1}{2^7}$, то общая вероятность нахождения правильных пар текстов для заданного трехраундового дифференциала составит $\left(\frac{1}{2^7}\right)^{18} = \frac{1}{2^{126}}$.

3. Разработка и реализация метода нахождения правильных пар текстов для дифференциального анализа алгоритма Кузнечик на примере. Так как вероятность нахождения правильных пар текстов достаточна маленькая, то быстро найти правильную пару текстов перебором всех значений с использованием обычного персонального компьютера не представляется возможным. Поэтому, для подтверждения действенности использования предложенного трехраундового дифференциала, был предложен способ «от обратного». Т.е. способ, позволяющий подобрать такие тексты, которые при определенных значениях секретного ключа будут образовывать нужные нам дифференциалы.

Первый раунд: на вход преобразования L подается один заполненный байт из 16-ти. Через преобразование L пройдет 16 различных вариантов заполнения каждого возможного байта от 00 до FF. Каждое однобайтовое значение раскладывается с помощью преобразования L на значение, состоящее из 16-ти байтов. В итоге на вход преобразования S второго раунда поочередно поступает 4096 128-битных значений. Чтобы найти все существующие пары значений нам нужно выполнить аналогичные преобразования во втором раунде.

Второй раунд: Аналогично берутся однобайтовые значения и проходят через преобразование L^{inv}, раскладываясь, таким образом, на 4096 128-битных значений.

В результате мы имеем дифференциалы ΔA и ΔC на входе и выходе преобразования S второго раунда соответственно. Так как для каждого значения ΔA не все значения ΔC равновероятны, нам нужно определить, какие именно поступающие на вход блока подстановки S дифференциалы ΔA из 4096 значений будут иметь ненулевую вероятность быть полученными в виде значений ΔC в результате пре-

образования. Для этого необходимо каждое из 4096 значений ΔA разбить на 16 байт и подать на вход таблицы вероятностей. Аналогично разбить на байты каждое значение ΔC. Для некоторых 16-ти байтных дифференциалов ΔA из 4096 значений по таблице найдутся такие дифференциалы ΔC, которые будут иметь ненулевую вероятность быть полученными.

В табл. 2 показан результат нахождения таких значений. Всего было найдено 13 пар значений ΔA/ΔC. Из этих значений можно вычислить, какие значения дифференциалов были поданы на вход алгоритма шифрования, а так же какие будут получены на выходе. Но для практического использования нам необходимо найти не дифференциалы текстов, а сами исходные тексты X и X', а также соответствующие им шифртексты.

Рассмотрим нахождение пар текстов, из которых состоят найденные дифференциалы на примере первой пары значений ΔA / ΔC:

ΔA = f3ab8c55c1099996fc5a4f2381976846

ΔC = 51ac91f0df24700190ad86a256131163.

Допустим, мастер-ключ

K=8899aabbccddeeff0011223344556677fedcba98765432100123456789abcdef.

Тогда на его основе были выработаны следующие 3 раундовых ключа:

- K1 = 8899aabbccddeeff0011223344556677;
- K2 = fedcba98765432100123456789abcdef;
- K3 = db31485315694343228d6aef8cc78c44.

В первом раунде нужно найти, из каких байтов было составлено значение ΔA. Эти байты можно найти из таблицы вероятностей, потому что ΔA = X⊕X'. Все возможные байты, из которых состоит значение ΔA, были найдены и приведены в табл. 3.

Таблица 2

Найденные пары значений *ΔA / ΔC*

ΔA	ΔC
f3ab8c55c199996fc5a4f2381976846	51ac91f0df2470190ad86a256131163
1a76bc71665284b01a3e595982599369	ba5a9d5e6d2b6431ac6b9cb72dc5a7a1
5cfbaa318fd91c774940bef22a5f86	1f8355405427f8e7d8c71cc07f2288c6
ac8ea817121caa3445efd4b9c43e875f	c6e355f95177d1bd58f9a4145283a143
522cbe6cbcf88eaf4963f28f8f29e62	353cceb5eab273db5790cc909cfa9
2bc22a75e57cbd804b35bf31ee5167	54be1b26f4dbe5b1f6a2a66a61e384d
5f28ebaf31b588b3f8f23923e399f0ef	b4eba9c9151b1fbd907f4f4d419fcdaa
7dbda6246e653ba46ec427fafa9a462f	5d87c43030f77ec08a9f25c0b8413318
efd9d1a17499185749edf1d1d6ee4c	bbd9f4a6c11bf5e154a2c52495e1ddd9
8fb8e26a91ccf9b72d6d5dce4f9ad4a9	b86cc61926ba16e11d19dce66e78450
6a7c998e18379bf720d423721d3c7e63	c6fc783ae4466b402df56e30dff1f8d4
337ddbfefc424a38d45ae559c2d2cd36	fd58a4739c68ed296855b6723e9fb3
5c9ce97665a2afd9172a54a881949c	3a3d1b8c2658ea7f8a958b2f866ecb5b

Количество всех возможных вариантов как для *X*, так и для *X'*: P = 2*2*4*4*2*2*2*2*2*2*2*2*2*2*4*2*2 = 524288.

Были найдены тексты, удовлетворяющие заданным условиям. В сумме каждая пара таких текстов даст значение выбранного дифференциала ΔA = f3ab8c55c1099996fc5a4f2381976846. Если каждое значение X и X' сложить с раундовым ключом K2 = fedcba98765432100123456789abcdef, то мы получим вход в преобразование S второго раунда. Например, X⊕K2 = 713113004cc561785706294165423220⊕ fedcba98765432100123456789abcdef = 8FEDA9983A91536856256C26ECE9FFCF, а X'⊕K2=829a9f558dccf8eeab5c6662e4d55a66⊕fedcba98765432100123456789abcdef = 7C4625CDFB98CAFEAA7F23056D7E9789.

Таблица 3

Байты, из которых состоит значение ΔA = f3ab8c55c1099996fc5a4f2381976846

Байт	**X**				**X'**			
f3	71		82		82		71	
ab	31		9a		9a		31	
8c	13	3d	9f	b1	3f	b1	13	3d
55	00	55	96	c3	55	00	c3	96
c1	4c		8d		8d		4c	
09	c5		cc		cc		c5	
99	61		f8		f8		61	
96	78		ee		ee		78	
fc	57		ab		ab		57	
5a	06		5c		5c		06	
4f	29		66		66		29	
23	41		62		62		41	
81	65		e4		e4		65	
97	42	7c	d5	eb	d5	eb	42	7c
68	32		5a		5a		32	
46	20		66		66		20	

Чтобы вернуться к предыдущему шагу – результату преобразования S первого раунда, нужно пропустить полученные тексты через L^{inv}. Значения до преобразования L^{inv} и после преобразования L^{inv} представлены в табл. 4.

Таблица 4

Пары текстов до и после преобразования L^{inv}

До L^{inv}	После L^{inv}
713113004cc561785706294165423220	18274d719b5138548224dc5fe0179f32
829a9f558dccf8eeab5c6662e4d55a66	9b31985c0bed35e5c995d28c289f31c4
713113004cc561785706294165423266	75a03d9bd2571480a87f4b0064d08424
829a9f558dccf8eeab5c6662e4d55a20	b0a915da436c6c02a0cbc467b0551e2b

Далее для получения исходных текстов нужно сделать преобразование S^{inv} с полученными значениями. Для получения текстов в результате S^{inv} нужно обратиться к таблице вероятностей, по которой можно определить наиболее вероятные тексты, на которые заменятся полученные ранее значения. Некоторые варианты значений, полученные в результате Sinv, показаны в табл. 5.

Теперь, для нахождения исходного текста, нужно сложить полученные значения с раундовым ключом K1. Ранее он был сформирован на основе заданного мастер ключа. Таким образом, если текст равен ce3110bf3555ecfeddd35b0d21080f,

а раундовый ключ равен 8899aabbccddeeff0011223344556677, то получаем исходный текст равный 46a8ba04f9880201feccf16849746e78. Таким образом, мы нашли исходный текст, который подавался на вход алгоритма шифрования. Для того, чтобы получить шифртекст по итогам трех раундов шифрования, нужно сделать следующие операции.

Таблица 5

Некоторые значения, полученные в результате S^{inv}, с помощью таблицы вероятностей

18274d719b5138548224dc5fe0179f32	9b31985c0bed35e5c995d28c289f31c4
ce3110bf3555ecfeddd35b0d21080f	f30d6013c7352d06781b16517b080df4
ce3110bf3555ecfeddd35b0d210832	f30d6013c7352d06781b16517b080d00

По аналогии нам нужно получить тексты, из которых состоит выходной дифференциал, взятый из табл. 2, например, значение

$$\Delta C = 51ac91f0df2470010x90ad86a256131163.$$

Были найдены значения текстов X и X', из которых состоит дифференциал ΔC. Далее нужно осуществить преобразование L второго раунда над полученными текстами. На рис. 5 приведены полученные значения в результате преобразования L.

```
c9e7794caa2ebdb7ca3888db3ca801a0
c9e7794caa2ebdb7ca6988db3ca801a0
822fcc6dab9177cbd1e945ce8f942885
822fcc6dab9177cbd1b845ce8f942885
ec2456a6f6faf4e23a88eb862c41ce71
ec2456a6f6faf4e23ad9eb862c41ce71
a7ece387f7453e9e215926939f7de754
a7ece387f7453e9e210826939f7de754
d7582800f74f784ed4f8b2e1ad92d57f
d7582800f74f784ed4a9b2e1ad92d57f
9c909d21f6f0b232cf297ff41eaefc5a
9c909d21f6f0b232cf787ff41eaefc5a
f29b07eaab9b311b2448d1bcbd7b1aae
f29b07eaab9b311b2419d1bcbd7b1aae
b953b2cbaa24fb673f991ca90e47338b
b953b2cbaa24fb673fc81ca90e47338b
```

Рис. 5. Некоторые значения, полученные в результате преобразования L второго раунда

Далее эти значения нужно сложить с третьим раундовым ключом и полученные значения подать на вход преобразования S третьего раунда. Завершающим этапом является выполнение преобразования L над полученными значениями. В результате будут получены правильные пары текстов, которые подтверждают работоспособность предложенной схемы.

Выводы. В данной статье описан алгоритм шифрования Кузнечик, для него разработаны и реализованы программно-ориентированные алгоритмы для выполнения зашифрования и расшифрования данных, а также алгоритм подбора правильных пар текстов в рамках дифференциального криптоанализа трех раундов шифрования. В ходе работы был предложен способ трехраундового построения характеристик для алгоритма шифрования Кузнечик, а также метод нахождения правильных пар текстов.

В результате выполнения исследований получен рабочий алгоритм шифрования Кузнечик, с помощью описанного метода найдены правильные пары текстов для дифференциального анализа 3-х раундов шифрования и использованием алгоритма Кузнечик. Таким образом, показано, что предложенная трехраундовая схема анализа, несмотря на маленькую вероятность, может быть использована, что подтверждается правильными парами текстами, искусственно подобранными (смоделированными) в лабораторных условиях.

Дальнейшие исследования в данной области будут состоять в продолжении анализа алгоритма Кузнечик. Будут осуществляться исследования по поиску и реализации оптимизированного алгоритма для дифференциального анализа данного шифра с целью получения новых результатов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Кузнечик (шифр). – URL: https://ru.wikipedia.org/wiki/Кузнечик_(шифр) (дата обращения 29.04.2017).
2. Криптографическая защита информации Блочные шифры – ГОСТ Р 34.12–2015. – URL: https://www.tc26.ru/standard/gost/GOST R 3412-2015.pdf (дата обращения: 29.04.2017).
3. *Ищукова Е.А., Кошуцкий Р.А., Бабенко Л.К.* Разработка и реализация высокоскоростного шифрования с использованием алгоритма "Кузнечик" // Журнал Auditorium. – 2015. – Вып. № 4 (8). "Общие и комплексные проблемы технических и прикладных наук и отраслей народного хозяйства".
4. *Толоманенко Е. А.* Программная реализация шифра "Кузнечик" // Материалы IX Международной студенческой электронной научной конференции «Студенческий научный форум» – 2017. "Актуальные проблемы информационной безопасности".
5. Дифференциальный криптоанализ. – URL: https://ru.wikipedia.org/wiki/ Дифференциальный_криптоанализ (дата обращения 29.04.2017).
6. *Бабенко Л.К., Ищукова Е.А.* Анализ симметричных криптосистем // Известия ЮФУ. Технические науки. – 2012. – № 12 (137). – С. 136-147.
7. *Бабенко Л.К., Ищукова Е.А.* Современные алгоритмы шифрования и методы их анализа: учеб. пособие. – М.: Гелиос АРВ, 2006. – 376 с.
8. *Biham E., Shamir A.*, Differential Cryptanalysis of the Full 16-round DES, Crypto'92, Springer-Velgar, 1998. – 487 p.
9. *Biham E., Shamir A.*, Differential Cryptanalysis of DES-like Cryptosystems, Extended Abstract, Crypto'90, Springer-Velgar, 1998. – 2 p.
10. *Бабенко Л.К. Ищукова Е.А.* Дифференциальный криптоанализ блочных шифров с применением распределенных вычислений // Материалы Международной научно-технической конференции «Многопроцессорные вычислительные и управляющие системы – 2007». Т. 1 – Таганрог: Изд-во ТТИ ЮФУ, 2007. – С. 222 -227.
11. *Ищукова Е.А., Бабенко Л.К.* Поиск дифференциалов с максимальными вероятностями // Проблемы информатизации общества. – Нальчик: Изд-во КБНЦ РАН, 2008. – С. 115-120.
12. *Babenko L.K., Ishchukova E.A.* Differential Analysis GOST Encryption Algorithm // Proceedings of the 3rd Internaternational Conference of Security of Information and Networks (SIN 2010). – P. 149-157, ACM, New York, 2010.
13. *Бабенко Л.К., Ищукова Е.А.* Анализ современных криптографических систем с помощью метода дифференциального криптоанализа // Актуальные аспекты защиты информации в Южном федеральном университете: монография. – Таганрог: Изд-во ТТИ ЮФУ, 2011. – С. 102-181.
14. *Бабенко Л.К., Ищукова Е.А.* Учебное пособие по курсу "Криптографические методы и средства обеспечения информационной безопасности". – Таганрог: Изд-во ТТИ ЮФУ, 2011. – 148 с.
15. *Панасенко С.* Алгоритмы шифрования. Специальный справочник. – СПб.: БХВ-Петербург, 2009. – 576 с.
16. *Шнайер Б.* Прикладная криптография: Протоколы, алгоритмы, исходные тексты на языке Си. – М.: ТРИУМФ, 2002. – 648 с.
17. *Столлингс В*., Криптография и защита сетей: принципы и практика. – 2-е изд.: пер. с англ. – М.: Изд. дом «Вильямс», 2001.

18. *Бабенко Л.К. Мишустина (Ищукова) Е.А.* Применение методов криптоанализа для исследования стойкости современных блочных шифров // Тезисы докладов X всероссийской научной конференции “Проблемы информационной безопасности в системе высшей школы”. – М.: МИФИ, 2003.
19. В ГОСТе сидел «Кузнечик». – URL: https://habrahabr.ru/post/266359/ (дата обращения 29.04.2017).
20. *Ищукова Е.А., Калмыков И.А.* Дифференциальные свойства S-блоков замены для алгоритма ГОСТ 28147-89 // Инженерный вестник Дона. – 2015. – № 4. –http://www.ivdon.ru/ru/magazine/archive/n4y2015/3284.

REFERENCES

1. Kuznechik (shifr) [Grasshopper (cipher)]. Available at: https://ru.wikipedia.org/wiki/Kuznechik_(shifr) (Accessed 29 April 2017).
2. Kriptograficheskaya zashchita informatsii Blochnye shifry – GOST R 34.12 – 2015 [Cryptography, Block ciphers, GOST R 34.12 – 2015]. Available at: https://www.tc26.ru/standard/gost/GOST R 3412-2015.pdf (Accessed 29 April 2017).
3. *Ishchukova E.A., Koshutskiy R.A., Babenko L.K.* Razrabotka i realizatsiya vysokoskorostnogo shifrovaniya s ispol'zovaniem algoritma "Kuznechik" [Development and implementation of high speed encryption algorithm "Grasshopper"], *Zhurnal Auditorium* [Auditorium], 2015, Issue No. 4 (8). "*Obshchie i kompleksnye problemy tekhnicheskikh i prikladnykh nauk i otrasley narodnogo khozyaystva"* ["Common and complex problems of technical and applied Sciences and branches national economy"].
4. *Tolomanenko E.A.* Programmnaya realizatsiya shifra "Kuznechik" [Software implementation of the cipher "Grasshopper"], *Materialy IX Mezhdunarodnoy studencheskoy elektronnoy nauchnoy konferentsii «Studencheskiy nauchnyy forum» – 2017. "Aktual'nye problemy informatsionnoy bezopasnosti"* [Materials of the IX International student electronic scientific conference "Student scientific forum" in 2017. "Actual problems of information security"].
5. Differentsial'nyy kriptoanaliz [The differential cryptanalysis]. Available at: https://ru.wikipedia.org/wiki/Differentsial'nyy_kriptoanaliz (Accessed 29 April 2017).
6. *Babenko L.K., Ishchukova E.A.* Analiz simmetrichnykh kriptosistem [Analysis of simmetric cryptosystems], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2012, No. 11 (136), pp. 136-147.
7. *Babenko L.K., Ishchukova E.A.* Sovremennye algoritmy shifrovaniya i metody ikh analiza: ucheb. posobie [Modern cryptographic algorithms and methods of analysis: textbook]. Moscow: Gelios ARV, 2006, 376 p.
8. *Biham E., Shamir A.*, Differential Cryptanalysis of the Full 16-round DES, Crypto'92, Springer-Velgar, 1998, 487 p.
9. *Biham E., Shamir A.*, Differential Cryptanalysis of DES-like Cryptosystems, Extended Abstract, Crypto'90, Springer-Velgar, 1998, 2 p.
10. *Babenko L.K. Ishchukova E.A.* Differentsial'nyy kriptoanaliz blochnykh shifrov s primeneniem raspredelennykh vychisleniy [Differential cryptanalysis of block ciphers with the use of distributed computing], *Materialy Mezhdunarodnoy nauchno-tekhnicheskoy konferentsii «Mnogoprotsessornye vychislitel'nye i upravlyayushchie sistemy – 2007»* [Materials of International scientific-technical conference "Multiprocessor computing and control systems – 2007"]. Vol. 1. Taganrog: Izd-vo TTI YuFU, 2007. pp. 222 -227.
11. *Ishchukova E.A., Babenko L.K.* Poisk differentsialov s maksimal'nymi veroyatnostyami [Search of differentials with maximum probability], *Problemy informatizatsii obshchestva* [Problems of Informatization of society]. Nal'chik: Izd-vo KBNTs RAN, 2008, pp. 115-120.
12. *Babenko L.K., Ishchukova E.A.* Differential Analysis GOST Encryption Algorithm, *Proceedings of the 3rd Internaternational Conference of Security of Information and Networks (SIN 2010)*, pp. 149-157, ACM, New York, 2010.
13. *Babenko L.K., Ishchukova E.A.* Analiz sovremennykh kriptograficheskikh sistem s pomoshch'yu metoda differentsial'nogo kriptoanaliza [Analysis of modern cryptographic systems using the method of differential cryptanalysis], *Aktual'nye aspekty zashchity informatsii v Yuzhnom federal'nom universitete: monografiya* [Actual aspects of information security in the southern Federal University: monograph]. Taganrog: Izd-vo TTI YuFU, 2011, pp. 102-181.

14. *Babenko L.K., Ishchukova E.A.* Uchebnoe posobie po kursu "Kriptograficheskie metody i sredstva obespecheniya informatsionnoy bezopasnosti" [The textbook for the course "Cryptographic methods and means of ensuring information security"]. Taganrog: Izd-vo TTI YuFU, 2011, 148 p.
15. *Panasenko S.* Algoritmy shifrovaniya. Spetsial'nyy spravochnik [The encryption algorithms. A special Handbook]. Saint-Petersburg: BKhV-Peterburg, 2009, 576 p.
16. *Shnayer B.* Prikladnaya kriptografiya: Protokoly, algoritmy, iskhodnye teksty na yazyke Si [Applied cryptography: Protocols, algorithms, and source code in C language]. Moscow: TRIUMF, 2002, 648 p.
17. *Stollings V.* Kriptografiya i zashchita setey: printsipy i praktika [Cryptography and network security: principles and practice]. 2nd ed.: The translation from English. Moscow: Izd. dom «Vil'yams», 2001.
18. *Babenko L.K. Mishustina (Ishchukova) E.A.* Primenenie metodov kriptoanaliza dlya issledovaniya stoykosti sovremennykh blochnykh shifrov [Application of methods of cryptanalysis to study the life of modern block ciphers], *Tezisy dokladov X vserossiyskoy nauchnoy konferentsii "Problemy informatsionnoy bezopasnosti v sisteme vysshey shkoly"* [Abstracts of the X all-Russian scientific conference "Problems of information security in higher school"]. Moscow: MIFI, 2003.
19. V GOSTe sidel «Kuznechik» [Guest sat "Grasshopper"]. Available at: https://habrahabr.ru/post/266359/ (Accessed 29 April 2017).
20. *Ishchukova E.A., Kalmykov I.A.* Differentsial'nye svoystva S-blokov zameny dlya algo-ritma GOST 28147-89 [Differential properties of S-block replacement for algo-rhythm GOST 28147-89], *Inzhenernyy vestnik Dona* [Engineering journal of Don], 2015, No. 4. Available at: http://www.ivdon.ru/ru/magazine/archive/n4y2015/3284.

Статью рекомендовал к опубликованию д.т.н., профессор Я.Е. Ромм.

Бабенко Людмила Климентьевна – Южный федеральный университет; e-mail: blk@fib.tsure.ru; 347928, г. Таганрог, ул. Чехова, 2, корпус "И"; тел.: 88634312018; кафедра безопасности информационных технологий; профессор.

Ищукова Евгения Александровна – e-mail: jekky82@mail.ru; тел.: 88634371905; кафедра безопасности информационных технологий; доцент.

Толоманенко Екатерина Алексеевна – e-mail: kat.tea@mail.ru; тел.: 88634371905; кафедра безопасности информационных технологий; студентка.

Babenko Lyudmila Klimentevna – Southern Federal University; e-mail: blk@fib.tsure.ru; Block "I", 2, Chehov street, Taganrog, 347928, Russia; phone: +78634312018; the department of security of information technologies; professor.

Ishchukova Evgeniya Aleksandrovna – e-mail: jekky82@mail.ru; phone: +78634371905; the department of security of information technologies; associate professor.

Tolomanenko Ekaterina Alekseevna – e-mail: kat.tea@mail.ru; phone: +78634371905; the department of security of information technologies; student.