УДК 681.3

ОТРИЦАЕМОЕ ШИФРОВАНИЕ НА ОСНОВЕ БЛОЧНЫХ ШИФРОВ

Н. А. Молдовян[а], *доктор техн. наук, заведующий лабораторией*
А. Р. Биричевский[а], *аспирант*
Я. А. Мондикова[б], *аспирантка*
[а]*Санкт-Петербургский институт информатики и автоматизации РАН, Санкт-Петербург, РФ*
[б]*Санкт-Петербургский государственный электротехнический университет «ЛЭТИ», Санкт-Петербург, РФ*

Постановка проблемы: *известные способы отрицаемого шифрования с разделяемым ключом, удовлетворяющие условию неотличимости по криптограмме от вероятностного шифрования, обладают сравнительно малой производительностью. Цель работы — повышение быстродействия алгоритмов отрицаемого шифрования, основанных на использовании блочных шифров.* ***Методы:*** *обращение блочного шифрующего преобразования, статистические эксперименты, одновременное шифрование двух независимых сообщений на двух различных ключах.* ***Результаты:*** *разработан новый способ выполнения процедуры отрицаемого шифрования, отличающийся от известных аналогов тем, что при зашифровании двух сообщений блочный шифр используется для выполнения прямого блочного преобразования по первому ключу и обратного преобразования по второму ключу. Получены формулы для оценки параметров алгоритмов на основе предложенного способа.* ***Практическая значимость:*** *предложенный способ представляет интерес для использования в средствах защиты информации от несанкционированного доступа.*

Ключевые слова *— компьютерная безопасность, криптография, отрицаемое шифрование, вероятностное шифрование, блочные шифры, криптограмма.*

Введение

В целях защиты информации от атак с принуждением предложены процедуры отрицаемого шифрования (ОШ) [1]. В атаках указанного типа предполагается, что атакующий перехватывает криптограмму и принуждает отправителя и получателя раскрыть исходное сообщение и использованные при шифровании ключи и случайные значения (если последние применялись). Различают ОШ по открытому ключу получателя сообщения и по разделяемому секретному ключу, которым владеют отправитель и получатель. При этом второй тип ОШ главным образом связывается с шифрованием двух и более сообщений на различных ключах и соединением независимых криптограмм в единый шифртекст. При принуждающей атаке предполагается достаточным предоставление атакующему только одного из ключей (фиктивного ключа) для расшифрования соответствующего ему участка шифртекста и восстановления фиктивного сообщения [1]. При этом оставшаяся часть шифртекста интерпретируется как случайная последовательность, использованная для маскирования криптограммы. Несмотря на явную наивность такого понимания ОШ по разделяемым секретным ключам, оно лежит в основе ряда средств компьютерной безопасности, например, Best Crypt [www.jetico.com/products/personal-privacy/bestcrypt-container-encryption], FreeOTFE [www.softpedia.com/get/Security/Encrypting/FreeOTFE.shtml], True Crypt [www.truecrypt.org].

Однако для защиты отдельных сообщений, которыми обмениваются удаленные пользователи, и отдельных файлов, хранимых в постоянной памяти ЭВМ, а также для построения защитных механизмов типа криптографических обманных ловушек [2], ориентированных на навязывание атакующему ложной информации, наивный подход к построению процедуры ОШ по разделяемому ключу представляется недостаточным. Действительно, наличие участков шифртекста, которые не используются для раскрытия фиктивного сообщения, дает атакующему существенные основания утверждать о неполноте раскрытия шифртекста. Устранение этого недостатка обеспечивается требованием неотличимости шифртекста, полученного в результате ОШ, от шифртекста, полученного в результате вероятностного шифрования фиктивного сообщения по фиктивному ключу, впервые обоснованным в работе [2] как одно из важных условий для обеспечения стойкости к принуждающим атакам.

В работе [3] описан общий способ построения алгоритмов ОШ, неотличимых по шифртексту от алгоритмов вероятностного шифрования, с использованием хэш-функций, а также представлен вариант реализации аналогичных алгоритмов с использованием блочных шифров. Однако недостатком способа [3] является сравнительно низкая производительность процедур ОШ, реализуемых на его основе.

В настоящей работе решается задача повышения производительности процедур ОШ, построенных на основе использования блочных шифров. Предложенный способ обеспечивает построение алгоритмов ОШ, обладающих в 100 и более раз высокой скоростью шифрования по сравнению с алгоритмами ОШ, построенными

на основе способа [3]. Существенный выигрыш в производительности достигается за счет того, что в процедуре ОШ блочный шифр используется не только в режиме шифрования, но и в режиме расшифрования.

Способ-прототип

Способ, предложенный в работе [3], позволяет любое трудно обратимое (однонаправленное) преобразование использовать для построения процедуры ОШ. В качестве однонаправленного преобразования рассматривались хэш-функции и блочные шифры. Отрицаемое шифрование предлагалось в виде одновременного шифрования двух сообщений (секретного и фиктивного) по двум независимым ключам, один из которых (фиктивный ключ) предназначен для раскрытия в случае принуждающей атаки. Способ легко расширяется на случай одновременного шифрования трех и более сообщений, однако при этом существенно падает производительность процедуры ОШ. Такое расширение интересно с теоретической точки зрения, но для практического применения его обоснование неочевидно, поэтому в дальнейшем будет рассмотрен только случай одновременного шифрования двух сообщений.

Важным требованием к процедурам ОШ является неотличимость криптограммы, формируемой в результате выполнения ОШ, от криптограммы, формируемой при вероятностном шифровании фиктивного сообщения. Это требование обеспечивается тем, что с алгоритмом ОШ ассоциируется некоторый алгоритм вероятностного шифрования (шифрования, в котором используются случайные значения) [2, 3]. Наряду с этим для последнего доказывается, что при определенных случайных значениях фиктивное сообщение в результате его шифрования по фиктивному ключу преобразуется в криптограмму, полученную в результате выполнения процедуры ОШ. При этом сама процедура ОШ также может быть как детерминистической, так и вероятностной. В последнем случае при выполнении процедуры ОШ используются случайные значения, которые определяют выработку конкретной криптограммы из множества возможных. В работе [3] предложен способ вероятностного ОШ, основанный на использовании односторонних преобразований, например хэш-функций.

В соответствии со способом [3] при использовании хэш-функции F_H процедура ОШ реализуется следующим образом. Пусть даны сообщения T (фиктивное) и M (секретное), представленные в виде последовательностей u-битовых знаков $\{t_1, t_2, ..., t_i, ..., t_z\}$ и $\{m_1, m_2, ..., m_i, ..., m_z\}$ соответственно. Одновременное шифрование этих сообщений по ключам K_T и K_M состоит в подборе таких случайных k-битовых значений $r_1, r_2, ..., r_i, ..., r_z$ ($k > 2u$), для которых одновременно выполняются соотношения

$$F_H(K_T, i, r_i) \bmod 2^u = t_i \text{ и } F_H(K_M, i, r_i) \bmod 2^u = m_i,$$

где предполагается использование хэш-функций, выходное значение которых имеет разрядность не менее u бит. Значение счетчика i включено в аргумент хэш-функции для улучшения статистических свойств криптограммы при сравнительно малых значениях k и u. Если последние два значения достаточно велики, то можно обойтись без задания зависимости значения хэш-функции от номера преобразуемого знака исходного текста.

В случае построения процедуры вероятностного ОШ на основе n-битового блочного шифра E ($n > 2u$) одновременное шифрование пары сообщений T и M по ключам K_M и K_T выполняется как подбор случайных значений r_i, удовлетворяющих следующим двум условиям:

$$E_{K_T}(r_i) \bmod 2^u = t_i \text{ и } E_{K_M}(r_i) \bmod 2^u = m_i. \quad (1)$$

Формируемая криптограмма имеет вид $R = \{r_1, r_2, ..., r_i, ..., r_z\}$. Расшифрование выполняется по формуле $E_K(r_i) \bmod 2^u = q_i$, где q_i — знаки восстановленного текста (T или M при $K = K_T$ или $K = K_M$ соответственно). При использовании блочных шифров обеспечивается более высокая скорость ОШ по сравнению со случаем использования хэш-функций. Однако по сравнению с производительностью $\lambda_{б.ш}$ [бит/с] блочного алгоритма шифрования E достигаемая скорость ОШ $\lambda_{ОШ}$ существенно меньше:

$$\lambda_{ОШ} \approx (2^{-2u-1}u/n)\lambda_{б.ш}. \quad (2)$$

Наиболее существенный вклад в снижение скорости шифрования при выполнении процедуры ОШ вносит необходимость многократного вычисления значений $E_{K_T}(r_i)$ и $E_{K_M}(r_i)$ для подбора такого r_i, при котором одновременно выполняются соотношения (1). Действительно, для текущего случайного значения r_i вероятность выполнения каждого из двух условий (1) равна 2^{-u}, а вероятность их одновременного выполнения — 2^{-2u}.

Существенное повышение производительности процедуры ОШ, основанной на выполнении шифрующих блочных преобразований, может быть достигнуто использованием свойства обратимости функции блочного шифрования E, которое принципиально отличает блочные шифры от хэш-функций. Однако для эксплуатации этого свойства требуется использовать в качестве знаков криптограммы выходные значения функции E, а не входные, как это имеет место в способе [3]. Предлагаемый новый вариант построения процедур ОШ на основе блочных шифров описывается в следующем разделе.

Новый способ отрицаемого шифрования

Предлагаемый способ ОШ реализуется как выполнение процедуры одновременного шифрования сообщений T и M в соответствии с формулами

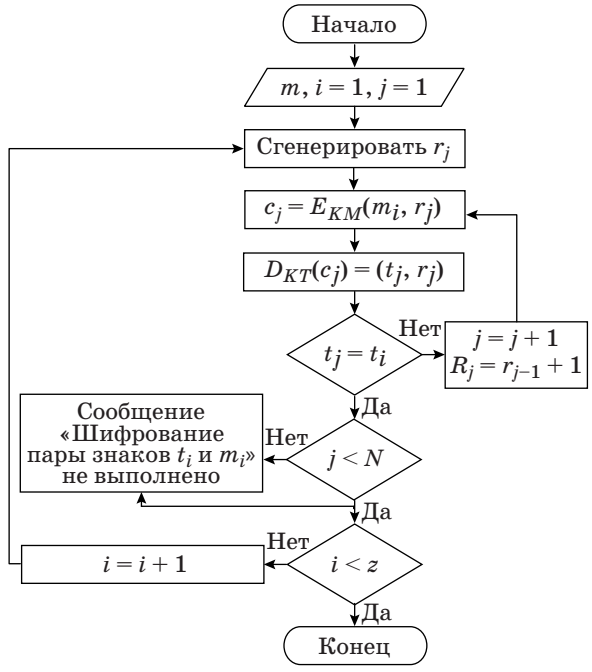
$$E_{K_T}(t_i, r_i') = c_i \text{ и } E_{K_M}(m_i, r_i) = c_i, \qquad (3)$$

где c_i — n-битовые блоки (знаки) криптограммы; r_i' и r_i — случайные значения. Восстановление знаков исходных сообщений из криптограммы $C = \{c_1, c_2, ..., c_i, ..., c_z\}$ предполагается осуществлять по формулам

$$D_{K_T}(c_i) = (t_i, r_i') \text{ и } D_{K_M}(c_i) = (m_i, r_i), \qquad (4)$$

где D — функция расшифрования, обратная к функции E, т. е. $D = E^{-1}$, а случайные k-битовые значения r_i' и r_i в выходном значении функции D отбрасываются ($k \geq 2u$; $n = u + k$), в результате чего получают знаки t_i и m_i. В соответствии с (3) процедура ОШ требует нахождения двух случайных значений r_i' и r_i, которые обеспечивают выполнимость условия $E_{K_T}(t_i, r_i') = E_{K_M}(m_i, r_i)$. Для случайно выбранных значений r_i' и r_i вероятность выполнения последнего равенства имеет достаточно малое значение $2^{-n} << 2^{-2u}$, что ограничивает скорость ОШ при использовании непосредственного подбора пар случайных значений r_i' и r_i.

Существенный выигрыш в производительности ОШ достигается применением следующего варианта нахождения блока криптограммы c_i по значениям t_i и m_i, обеспечивающего выполнение условий (3) и лежащего в основе алгоритма 1 (рис. 1). Предварительно выбирается произвольное значение r_j и вычисляется значение $c_j = E_{K_M}(m_i, r_j)$, затем — значение $D_{K_T}(c_j)$, которое рассматривается как конкатенация u-битового значения t_j и k-битового значения r_j', т. е. $D_{K_T}(c_j) = (t_j, r_j')$. При нахождении такого значения r_j, при котором $t_j = t_i$, полученное значение c_j может быть взято в качестве блока криптограммы c_i. При этом вероятность выполнения равенства $t_j = t_i$ (совпадение случайного u-битового значения t_j с заданным u-битовым значением t_i) равна $2^{-u} >> 2^{-2u}$, что определяет существенное повышение скорости ОШ по сравнению со способом [3].



■ ***Рис. 1.*** Блок-схема разработанного алгоритма ОШ

Алгоритм 1: совместное шифрование сообщений T и M.

1. Установить значение счетчика $i = 1$.
2. Установить значение счетчика $j = 1$.
3. Сгенерировать случайное k-битовое число r_j.
4. Вычислить значение $c_j = E_{K_M}(m_i, r_j)$.
5. Вычислить значение $D_{K_T}(c_j) = (t_j, r_j')$, где выходное n-битовое значение функции расшифрования интерпретируется как конкатенация u-битового значения t_j и k-битового значения r_j'.
6. Сравнить значения t_j и t_i. Если $t_j = t_i$, то взять в качестве значения c_i значение c_j и перейти к шагу 7, в противном случае перейти к шагу 3.
7. Если $j < N$, то прирастить значение счетчика $j \leftarrow j + 1$, вычислить $r_j \leftarrow r_{j-1} + 1$ и перейти к шагу 4, иначе вывести сообщение «Шифрование пары знаков t_i и m_i не выполнено». (Вопрос о выборе значения N рассмотрен в следующем разделе.)
8. Если $i < z$, то прирастить значение счетчика $i \leftarrow i + 1$ и перейти к шагу 2, иначе СТОП.

Алгоритм 1, описывающий процедуру ОШ, формирует выходную криптограмму C, которая вычислительно неотличима от криптограммы, полученной в процессе следующего алгоритма вероятностного шифрования фиктивного сообщения. Данный алгоритм относится к вероятностным процедурам ОШ, в которых в ходе шифрования используются случайные значения.

Алгоритм 2: ассоциируемый алгоритм вероятностного шифрования фиктивного сообщения M по фиктивному ключу K_M.

1. Установить значение счетчика $i = 1$.
2. Сгенерировать случайное k-битовое число r_i.
3. Вычислить значение $c_i = E_{K_M}(m_i, r_i)$.
4. Если $i < z$, то прирастить значение счетчика $i \leftarrow i + 1$ и перейти к шагу 2, иначе СТОП.

Наличие алгоритма 2 явно показывает, что алгоритм 1 удовлетворяет требованию неотличимости ОШ от вероятностного шифрования. Действительно, потенциально реализуем выбор случайных значений r_i, при котором выходная криптограмма алгоритма 2 совпадает с выходной криптограммой алгоритма 1 (при одних и тех же значениях M и K_M).

Алгоритм 3: расшифрование криптограммы C по ключу K.

1. Установить ключ шифрования $K = K_M$ (восстановление фиктивного сообщения M) или $K = K_T$ (восстановление секретного сообщения T) и значение счетчика $i = 1$.

2. Вычислить значение $\chi_i = D_K(c_i) \operatorname{div} 2^k$.

3. Если $i < z$, то прирастить значение счетчика $i \leftarrow i + 1$ и перейти к шагу 2, иначе СТОП.

Алгоритм 3 выдает выходное сообщение $\{\chi_1, \chi_2, ..., \chi_i, ..., \chi_z\}$, которое совпадает с фиктивным сообщением M, если было установлено значение ключа $K = K_M$, или с секретным сообщением T, если было установлено значение ключа $K = K_T$.

Описанные алгоритмы 1, 2 и 3 реализуют передачу секретного сообщения T между удаленными пользователями, которые предварительно согласовывают общий секретный ключ K_T и общий фиктивный ключ K_M по следующему сценарию, стойкому к принуждающей атаке.

1. Алиса (отправитель) генерирует фиктивное сообщение M, после чего, используя алгоритм 1 и ключи K_T и K_M, зашифровывает совместно сообщения T и M и полученную криптограмму C направляет по открытому каналу Бобу (получателю).

2. Боб расшифровывает криптограмму C, используя алгоритм 3 и ключ K_T, и получает секретное сообщение T.

3. Ева (атакующий), имеющая возможность наблюдать за трафиком открытого канала, перехватывает криптограмму C, после чего принуждает одновременно Алису и Боба раскрыть сообщение, содержащееся в криптограмме C, и ключ шифрования.

4. Алиса (и Боб) предоставляют Еве ключ K_M и алгоритм 2 как алгоритм, использованный для зашифрования переданного (полученного) сообщения, и алгоритм 3 как алгоритм для расшифрования сообщения.

5. Ева расшифровывает криптограмму C, используя алгоритм 3 и ключ K_M, в результате чего получает сообщение M. После этого Ева зашифровывает сообщение M по ключу K_M в соответствии с алгоритмом 2, используя случайные k-битовые значения r_i, восстановленные при расшифровании криптограммы C, и убеждается, что сообщение M действительно преобразуется в криптограмму C.

Для того чтобы уличить Алису и Боба в обмане, Ева должна взломать базовый алгоритм блочного шифрования, т. е. вычислить ключ K_T, что практически невыполнимо, если разрядность этого ключа составляет 128 бит и более, а в качестве базового алгоритма используется стойкий блочный шифр, например ГОСТ 28147 [4].

Выбор параметров и оценка производительности алгоритма отрицаемого шифрования

Производительность алгоритма, описанного в предыдущем разделе, существенно зависит от среднего числа η выбираемых значений r_j при шифровании одной пары знаков t_i и m_i. Значение η зависит от вероятности выполнения на шаге 6 условия $t_j = t_i$, которая равна $\Pr(t_j = t_i) = 2^{-u}$ и определяется разрядностью знаков t_i и m_i. Вероятность невыполнения условия $t_j = t_i$ для μ последовательных значений $j = 1, 2, ..., \mu$ равна

$$\Pr_\mu(t_j \neq t_i) = (1 - 2^{-u})^\mu, \qquad (5)$$

откуда получаем значение вероятности выполнения условия $t_j = t_i$ на μ-м шаге

$$\Pr_\mu(t_j = t_i) = 1 - (1 - 2^{-u})^\mu \qquad (6)$$

и значение η:

$$\eta = 2^{-u} + 2[1 - (1 - 2^{-u})^2] + ... + \\ + \mu[1 - (1 - 2^{-u})^\mu] + ... + N[1 - (1 - 2^{-u})^N].$$

Более удобной является приближенная формула

$$\eta \approx [\Pr(t_j = t_i)]^{-1} = 2^u, \qquad (7)$$

с помощью которой получаем следующую оценку производительности предложенного алгоритма ОШ:

$$\lambda'_{\text{ОШ}} \approx (2^{-u-1}u/n)\lambda_{\text{б.ш}}. \qquad (8)$$

Сравнение с формулой (2), относящейся к ОШ по способу [3], показывает, что предложенный вариант реализации процедуры ОШ на основе блочных шифрующих преобразований дает увеличение производительности ОШ примерно в 2^u раз.

Значение N в предложенном алгоритме выбирается с учетом получения достаточно малого значения $\Pr_N(t_j \neq t_i)$, т. е. вероятности того, что шифрование текущей пары знаков t_i и m_i не будет выполнено. Из формулы (5) получаем уравнение для вычисления N по заданному значению $\Pr_N(t_j \neq t_i)$:

$$\Pr_N(t_j \neq t_i) = (1 - 2^{-u})^N . \qquad (9)$$

Оценим порядок некоторого значения $N' > N > \eta$, для чего подставим в (5) значение $\mu = \eta/2 \approx 2^{u-1}$ и воспользуемся соотношением

$$\Pr_\mu(t_j \neq t_i) < 1 - \mu 2^{-u} = 1 - 2^{u-1}2^{-u} = 1/2. \qquad (10)$$

Обозначая целую часть отношения $N'/(\eta 2^{-1})$ как некоторое натуральное число ω, получаем

$$\Pr_{N'}(t_j \neq t_i) < (1/2)^\omega = (1/2)^{2N'/\eta}. \qquad (11)$$

Задавая пороговое значение вероятности $\Pr_{N'}(t_j \neq t_i)$ в виде отрицательной степени s числа 2, легко вычислить N':

$$2^{-s} > 2^{-N'/\eta} \Rightarrow N' \geq s\eta = s2^{u-1}. \qquad (12)$$

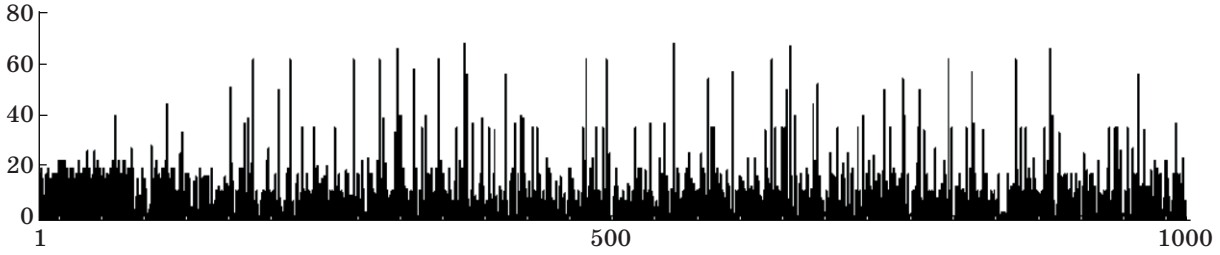
Последнее значение может быть использовано в качестве значения N в алгоритме 1, задающем процедуру ОШ. Выбор больших значений параметра N в алгоритме 1 практически не влияет на его производительность, поскольку случаи, когда требуется выполнить число попыток выбора значения r_j, существенно превосходящее значение η, встречаются сравнительно редко.

Типичная зависимость количества выборок значений r_j от номера шифруемого блока при значении параметра $u = 4$ показана на рис. 2. Из графика видно, что большинство значений лежит в пределах 20 выборок на блок. Среднее число η выборок значений r_j, определенное экспериментально для различных режимов ОШ, близко к теоретическим значениям, вычисляемым по формуле (7).

Значение параметров u и n следует выбирать таким образом, чтобы число N_r всех возможных случайных значений r превосходило N, для чего достаточно выполнения неравенства $N_r > N'$. Поскольку $N_r = 2^k$, последнее неравенство выполняется при $2^k > 2^u \log_2 \Pr_{N'}(t_j \neq t_i)$. Варианты выбора параметров алгоритма ОШ, представляющие практический интерес, приведены в табл. 1.

Экспериментальная проверка эффективности работы разработанного способа отрицаемого шифрования была выполнена на базе программной платформы Microsoft .NET Framework. Результаты измерений скорости шифрования приведены в табл. 2.

Производительность предложенного способа ОШ прямо пропорциональна производительности используемого алгоритма блочного шифрования. Современные аппаратные реализации алгоритма ГОСТ 28147 обеспечивают производительность 1000 Мбит/с и более. При таких реализациях



■ ***Рис. 2.*** Зависимость количества выборок значений r_j от значения i при использовании алгоритма ГОСТ 28147 в качестве базового блочного шифра и параметров $u = 4$, $k = 60$

■ ***Таблица 1.*** Варианты выбора параметров алгоритма ОШ и оценка достигаемой скорости шифрования ($\lambda'_{ОШ}$)

n	u	k	η	$\Pr_{N'}(t_j \neq t_i)$ (s)	N'	N_r	$\lambda'_{ОШ}$, отн. ед.
32	4	28	2^4	2^{-16} (2^4)	2^7	2^{28}	2^{24}
32	8	24	2^8	2^{-16} (2^4)	2^{11}	2^{24}	2^{21}
64	8	56	2^8	2^{-32} (2^5)	2^{12}	2^{56}	2^{20}
64	8	56	2^8	2^{-64} (2^6)	2^{13}	2^{56}	2^{20}
96	8	88	2^8	2^{-32} (2^5)	2^{12}	2^{88}	2^{20}
96	12	84	2^{12}	2^{-64} (2^6)	2^{17}	2^{84}	2^{16}
128	16	112	2^{16}	2^{-64} (2^6)	2^{21}	2^{112}	2^{12}

■ ***Таблица 2.*** Результаты измерений скорости шифрования алгоритма ОШ при использовании в качестве базового алгоритма блочного шифрования шифров RC5 [5] и ГОСТ 28147 [4]

Режим работы алгоритма ОШ	Производительность алгоритма 1: эксперимент/формула (8), бит/с	Значение η: эксперимент/формула (7)	Производительность базового шифра, бит/с
$n = 32$, $u = 8$, $k = 24$ (бит); базовый шифр — RC5	1150/1360	257/256	2 728 211
$n = 32$, $u = 4$, $k = 28$ (бит); базовый шифр — RC5	10260/10920	14/16	2 728 211
$n = 64$, $u = 8$, $k = 56$ (бит); базовый шифр — ГОСТ 28147	1176/1000	240/256	4 093 953
$n = 64$, $u = 4$, $k = 60$ (бит); базовый шифр — ГОСТ 28147	9627/8000	15/16	4 093 953

производительность ОШ на базе ГОСТ 28147-89 будет достигать значения 6,3 Мбит/с.

Также ощутимое влияние на скорость ОШ оказывает значение параметра u. Однако повышение быстродействия за счет уменьшения значения u ограничивается тем, что при этом возрастает отношение размера криптограммы к размеру шифруемых сообщений. По сравнению со способом ОШ [3], основанным на использовании блочных шифров, алгоритм 1 обладает производительностью в $\psi \approx 2^u$ раз более высокой при использовании одного и того же базового блочного шифра. Значение коэффициента ψ равно 16 при $u = 4$ и 256 при $u = 8$.

Заключение

Впервые алгоритмы ОШ, удовлетворяющие требованию неотличимости от вероятностного шифрования, предложены в работе [2] с использованием операций возведения в большую дискретную степень по простому модулю. При таком подходе к построению алгоритмов ОШ требуется выполнение самостоятельных исследований их стойкости. В работе [3] впервые предложено построение алгоритмов ОШ указанного типа с использованием известных хэш-функций и алгоритмов блочного шифрования, стойкость которых хорошо исследована. Основной результат настоящей работы состоит в разработанном новом способе ОШ, удовлетворяющего требованию неотличимости от вероятностного шифрования и основанного на использовании блочных шифров. Предложенный способ позволяет построить алгоритмы ОШ, производительность которых существенного выше (в 16 раз и более) по сравнению с алгоритмами, основанными на способе [3]. Применение стойких блочных шифров для реализации предложенного способа ОШ обеспечивает стойкость разрабатываемых на его основе алгоритмов ОШ. Действительно, гипотетическая успешная атака на такие алгоритмы ОШ, например атака на основе известных или специально подобранных исходных текстов, может быть применена и для взлома базового алгоритма шифрования.

Другим результатом выполненной работы является получение формул для оценки значений параметров алгоритма ОШ, основанного на предложенном способе. Из данных табл. 1 видно, что производительность не зависит от выбираемого значения N, что объясняется малой вероятностью случаев, когда процесс шифрования пары знаков исходных тестов t_i и m_i потребует выполнения N циклов, включающих шаги 3–6 алгоритма ОШ, если задано малое значение вероятности $\Pr_N(t_j \neq t_i)$. Экспериментом подтверждено, что среднее число (η) выбираемых значений r_j много меньше значения N.

В качестве базового алгоритма блочного шифрования E представляет интерес использование 64-битовых скоростных шифров с высокой эффективностью аппаратной реализации [6–8], что позволит достигнуть производительности ОШ, равной 10–100 Мбит/с. Более высокая производительность достигается при использовании блочных шифров с малым размером входного блока n, однако использование значений $n < 32$ требует решения задачи разработки режимов использования ОШ, обеспечивающих улучшение маскирования статистических свойств исходных текстов. Последнее представляет самостоятельную задачу отдельного исследования.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 14-07-00061-а.

Литература

1. **Canetti R., Dwork C., Naor M., Ostrovsky R.** Deniable Encryption // Advances in Cryptology — CRYPTO 1997: Proc.17th Annual Intern. Cryptology Conf., Santa Barbara, California, USA, Aug. 17–21, 1997. P. 90–104.
2. **Березин А. Н., Биричевский А. Р., Молдовян Н. А., Рыжков А. В.** Способ отрицаемого шифрования // Вопросы защиты информации. 2013. № 2. С. 18–21.
3. **Морозова Е. В., Мондикова Я. А., Молдовян Н. А.** Способы отрицаемого шифрования с разделяемым ключом // Информационно-управляющие системы. 2013. № 6. С. 73–78.
4. **ГОСТ** 28147–89. Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования. — М.: Изд-во стандартов, 2007. — 28 с.
5. **Rivest R. L.** The RC5 Encryption Algorithm // Fast Software Encryption: Proc. 2nd Int. Workshop. 1995. Vol. 1008. P. 86–96.
6. **Moldovyan N. A., Moldovyan A. A.** Data-Driven Block Ciphers for Fast Telecommunication Systems. Auerbach Publications. — N. Y.; London: Talor & Francis Group, 2008. — 185 p.
7. **Moldovyan N. A., Moldovyan A. A., Eremeev M. A.** A Class of Data-Dependent Operations // International Journal of Network Security. 2006. Vol. 2. N 3. P. 187–204.
8. **Moldovyan N. A., Moldovyan A. A., Sklavos N.** Controlled Elements for Designing Ciphers Suitable to Efficient VLSI Implementation // Telecommunication Systems. 2006. Vol. 32. N 2/3. P. 149–163.

UDC 681.3

Deniable Encryption Based on Block Ciphers

Moldovyan N. A.[a], Dr. Sc., Tech., Head of a Research Laboratory, Professor, nmold@mail.ru
Birichevskiy A. R.[a], Post-Graduate Student, lehabirich@mail.ru
Mondikova Ya. A.[b], Post-Graduate Student, mondikovay@gmail.com
[a]Saint-Petersburg Institute for Informatics and Automation of RAS, 39, 14 Line, V. O., 199178, Saint-Petersburg, Russian Federation
[b]Saint-Petersburg State Electrotechnical University "LETI", 5, Professora Popova St., 197376, Saint-Petersburg, Russian Federation

Purpose: The available methods of shared-key deniable encryption satisfying the criterion of indistinguishability from probabilistic encryption have comparatively low performance. This paper is devoted to developing a method for faster deniable encryption based on using block ciphers. **Methods:** Reversing the block-ciphering transformation, statistic experiments, simultaneous encryption of two independent messages using two different keys. **Results:** A new method was developed to carry out deniable encryption by performing direct block transformation with the first key and inverse block transformation with the second key when encrypting two messages. The formulas were derived to estimate the parameters of the algorithms based on the proposed method. **Practical relevance:** The proposed method can be applied in computer security systems.

Keywords — Computer Security, Cryptography, Deniable Encryption, Probabilistic Encryption, Block Ciphers, Cryptogram

References

1. Canetti R., Dwork C., Naor M., Ostrovsky R. Deniable Encryption. *Advances in Cryptology — CRYPTO 1997*. Proc. 17th Annual Intern. Cryptology Conf., 1997, pp. 90–104.
2. Birichevskiy A. R., Moldovyan N. A., Berezin A. N., Ryzhkov A. V. A Method of the Deniable Encryption. *Voprosy zashchity informatsii*, 2009, no. 2, pp. 18–21 (In Russian).
3. Morozova E. V., Mondikova Y. A., Moldovyan N. A. Methods of Deniable Encryption with a Shared Key. *Informatsionno-upravliaiushchie sistemy*, 2013, no. 6, pp. 73–78 (In Russian).
4. Russian Standard GOST 28147–89. Systems for Processing Information. Cryptographic Protection. Algorithm for Cryptographic Transformation. Moscow, Standartov Publ., 2007. 28 p. (In Russian).
5. Rivest R. L. The RC5 Encryption Algorithm. *Proc. 2nd Int. Workshop "Fast Software Encryption"*, 1995, vol. 1008, pp. 86–96.
6. Moldovyan N. A., Moldovyan A. A. Data-driven Block Ciphers for Fast Telecommunication Systems. Auerbach Publications. New York, London, Talor & Francis Group, 2008. 185 p.
7. Moldovyan N. A., Moldovyan A. A., Eremeev M. A. A Class of Data-dependent Operations. *International Journal of Network Security*, 2006, vol. 2, no. 3, pp. 187–204.
8. Moldovyan N. A., Moldovyan A. A., Sklavos N. Controlled Elements for Designing Ciphers Suitable to Efficient VLSI Implementation. *Telecommunication Systems*, 2006, vol. 32, no. 2/3, pp. 149–163.