УДК 681.3.06

МИНИ-ВЕРСИЯ БЛОЧНОГО СИММЕТРИЧНОГО АЛГОРИТМА КРИПТОГРАФИЧЕСКОГО ПРЕОБРАЗОВАНИЯ ИНФОРМАЦИИ С ДИНАМИЧЕСКИ УПРАВЛЯЕМЫМИ КРИПТОПРИМИТИВАМИ (BABY-ADE)

В.И. ДОЛГОВ, А.А. КУЗНЕЦОВ, Р.В. СЕРГИЕНКО, А.Л. БЕЛОКОВАЛЕНКО

В первой части статьи приводится описание уменьшенной модели шифра ADE – Baby-ADE – одного из претендентов, заявленных на Украинский конкурс по выдвижению и отбору кандидатов на национальный стандарт блочного симметричного шифрования, а во второй рассматривается выполнение атаки невозможных дифференциалов и атаки «Квадрат» на уменьшенные модели шифров mini-AES и baby-ADE. На основе сравнительного анализа результатов проведения этих атак делается вывод, что введение в шифр ADE зависимости внутренних операций цикловых преобразований от ключа позволяют значительно повысить временную сложность атак, что приводит к повышению запаса стойкости (security margin) шифра.

The first part of the paper describes the diminished model of the ADE cipher – BabyAGE – one of the claimants to the Ukrainian competition on nomination and selection of candidates to the national standard of block symmetric ciphering and the second one considers the execution of the attack of impossible differentials and the attack «Kvadrat» on the reduced models of miniAES and babyADE ciphers. The paper concludes on the basis of comparative analysis of the results of executing these attacks that the introductions into the ADE cipher of dependence of inner operations of cycle transformations on the key allows to considerably increase temporary complexity of attacks, which results in the increase of the cipher's security margin.

ВВЕДЕНИЕ

В настоящее время в Украине проходит открытый конкурс по выдвижению и отбору кандидатов на национальный стандарт блочного симметричного шифрования. Одним из предложений, подготовленных с участием двух авторов настоящей статьи, является алгоритм криптографических преобразований информации с динамически управляемыми криптопримитивами, названный ADE (Algorithm of Dynamic Encryption) [13].

На текущем этапе конкурса проходит изучение предложений экспертами и специалистами, а также ведется работа по проверке заявленных показателей стойкости и производительности.

Конечно, выполнить всесторонний криптоанализ и проверку надежности шифра – это довольно не простая задача. Ее решение требует больших временны*х* и интеллектуальных ресурсов, а для некоторых видов криптоанализа оно практически невозможно [1, 2, 4, 5, 6–10, 13, 14, 15 и мн. др.].

В широком спектре стоящих задач большое значение приобретает развитие и применение технологий, позволяющих ускорить процессы исследования и принятия решений. Одним из таких путей, направленных на создание и отработку эффективных методов сопоставления различных предложений, может стать, на наш взгляд, анализ криптографических показателей уменьшенных версий (моделей) шифров, в которых сохранены все принципиальные преобразования основного (большого) шифра. Естественно, здесь сразу возникает вопрос об адекватности перехода к версиям малых шифров (в смысле сохранения в модели всех свойств прототипа). Однако здесь можно положиться на достаточно очевидный принцип (назовем его постулатом): если хороши свойства модели, то свойства прототипа как минимум будут не хуже. Когда прототип поддается масштабируемости, то есть удается в модели сохранить структуру преобразований блоков данных и свойства основных операций, то результаты анализа свойств модели при определенных условиях могут быть перенесены на прототип.

В порядке реализации этого подхода возникает самостоятельная важная задача построения уменьшенных моделей шифров с последующим их анализом и исследованием. Рассматриваемая задача имеет еще и определенное методическое значение в виде создания базы, которая может быть использована в учебных целях для изучения методов криптоанализа.

В этой работе предлагается уменьшенная модель шифра ADE – Baby-ADE,– одного из претендентов, заявленных на конкурс.

1. МОТИВАЦИЯ РАЗРАБОТКИ ADE

Алгоритм ADE был разработан на основе шифра AES, отобранного по результатам конкурса в качестве стандарта шифрования США и являющегося в настоящее время одним из лучших по показателям быстродействия и стойкости, а также по открытости и прозрачности всех механизмов преобразований [1, 2]. Однако в последнее время в открытой печати все чаще появляются публикации, в которых авторы подчеркивают потенциальную уязвимость шифра AES к атакам, которые могут использовать простоту алгебраического описания шифра [например, 14, 15 и др.].

Мы посчитали, что дополнительное повышение показателей стойкости шифра AES может быть достигнуто на основе введения в шифрую-

щие преобразования механизмов динамического управления промежуточными состояниями. Они позволяют получить дополнительное повышения сложности (размерности) системы алгебраических уравнений, описывающих процедуру зашифрования, не затрагивая принципиальной основы использованных в AES решений. С этой целью в структуру цикловых преобразований, включающих в AES операции рассеивания, сдвига и нелинейной побайтной замены, введены ключезависимые параметры, позволяющие реализовать динамическое изменение результатов каждой из операций в зависимости от текущего значения ключевых битов. Эффект введения динамики отражен и в названии шифра – Алгоритм Динамического Шифрования.

Все дополнительно введенные преобразования и операции мы постарались сохранить и в уменьшенной модели ADE. Материал, излагаемый далее, посвящается детальному описанию этой уменьшенной модели. При изложении материала мы в значительной степени будем опираться на описание шифра mini-AES, опубликованное в открытой печати [9].

2. BABY-ADE

Как и в mini-AES, в нашем шифре Baby-ADE исходное входное сообщение, названное открытым текстом, разбивается на блоки по 16 бит каждый. За одно применение алгоритма осуществляется шифрование только одного блока открытого текста. Процесс повторяется до тех пор, пока не будут зашифрованы все блоки. Для зашифрования используется ключ длиной 16 бит (как и mini-AES). В процессе зашифрования шифртекст составляется из зашифрованных блоков, последовательность которых соответствует очередности блоков открытого текста.

Мы посчитали необходимым взять большее число цикловых преобразований, чем их предусмотрено в mini-AES. Их число в соответствии с решаемыми прикладными задачами исследования может принимать значения 4 или 8. Цикловые преобразования идентичны по структуре, поэтому изменение количества циклов существенно не повлияет на общее описание шифра, так же как и на описание процедуры расширения ключа.

Рис. 1, заимствованный из описания mini-AES [9], иллюстрирует процесс шифрования сообщения с помощью Baby-ADE.

Рассмотрим особенности реализации отдельных компонент шифра Baby-ADE.

2.1 Компоненты шифра Baby-ADE

Сходство mini-AES и Baby-ADE позволяет во многих случаях воспользоваться иллюстративными материалами и концепцией описания, представленными в спецификации шифра mini-AES.

Для простоты описания внутренних процессов шифрования Baby-ADE входной 16-битный блок открытого текста p, состоящий из последовательности полубайтов p_0, p_1, p_2, p_3 или вектор-строки $p = (p_0, p_1, p_2, p_3)$, представляется в виде матрицы размера 2×2, названной в соответствии с терминологией, использованной при описании шифра AES состоянием. Отмеченный подход иллюстрирует рис. 2.

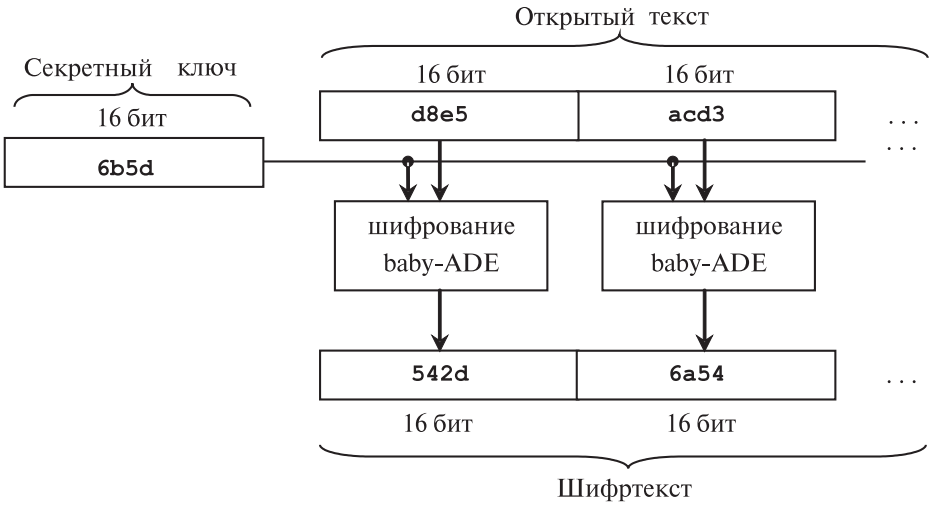

Рис. 1. Шифрование открытого сообщения шифром Baby-ADE

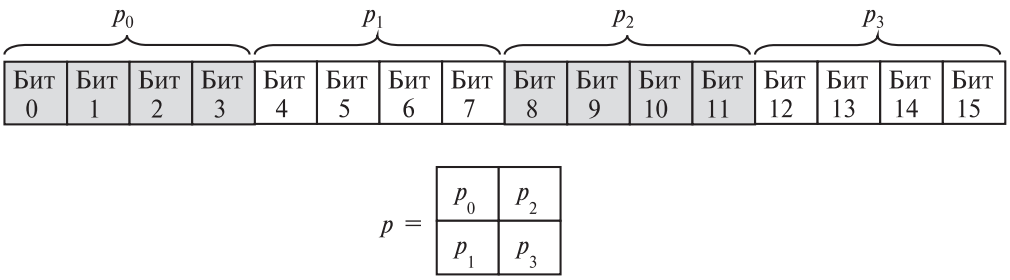

Рис. 2. Представление 16-битного блока в виде матрицы 2×2

Для представления полубайтов наряду с двоичной формой в дальнейшем будет использоваться и шестнадцатеричная форма представления (как это уже сделано на рис. 1).

Например, если входной блок 1000 1100 0111 0001, состоит из полубайтов $p_0 = 8$, $p_1 = c$, $p_2 = 7$, $p_3 = 1$, то соответствующая матрица состояния будет иметь вид

$$\begin{bmatrix} 8 & 7 \\ c & 1 \end{bmatrix}, \text{ или } \begin{bmatrix} 1000 & 0111 \\ 1100 & 0001 \end{bmatrix}.$$

Подобным же образом представляется и секретный ключ – как 4 полубайта $k_0k_1k_2k_3$ и соответствующее им состояние $\begin{bmatrix} k_0 & k_2 \\ k_1 & k_3 \end{bmatrix}$.

В процессе зашифрования в ADE, как и в шифре AES, принимают участие 4 основных компоненты, а именно операции *SubBytes*, *ShiftRow*, *MixColumn* и *KeyAddition*, последовательное применение которых и составляет содержание циклового преобразования. Введенные в шифр ADE изменения коснулись содержания всех этих внутренних компонентов, – они стали управляемыми ключевыми битами. Указанные названия основных преобразований сохранены и в шифрах mini-AES и Baby-ADE, за исключением преобразования SubBytes. В шифре mini-AES процедуре SubBytes соответствует полубайтовое преобразование NibbleSub. Оно является простой операцией, которая заменяет каждый входной полубайт выходным полубайтом в соответствии с фиксированной таблицей подстановок. В качестве подстановки используется первая строка первого S-блока DES. В шифре Baby-ADE соответствующее преобразование названо SubHalfBytes.

2.2 SubHalf-Bytes, γ

Операцию SubHalfBytes иллюстрирует рис. 3.

Рис. 3. Операция SubHalfBytes

В самом шифре AES соответствующая процедура задается аффинным преобразованием вида

$$b = M \cdot (a)^{-1} + \beta,$$

где M – квадратная невырожденная матрица размером 8×8 с элементами из поля $\mathbf{GF}(2)$, a и b – 8-и битные векторы значений входа и выхода преобразования SubBytes соответственно (элементы соответствующих матриц-состояний), а β – фиксированный 8-и битный вектор, являющийся заданным параметром этого преобразования (т.е. $a, b, \beta \in \mathbf{GF}(2^8)$).

В отличие от AES, в шифре ADE используются изменяемые таблицы блоков замены, формируемые с помощью дополнительно введенного параметра $\gamma \in \mathbf{GF}(2^8)$, который определяется битами расширенного ключа. Мы повторили идею этого преобразования и в шифре Baby-ADE, только она отмасштабирована соответственно размеру 16-битного состояния.

В результате в качестве *S*-блока выступает изменяемая матрица подстановок, которая строится с помощью вычисления мультипликативно обратного элемента $(a \cdot \gamma)^{-1} \in \mathbf{GF}(2^4)$ с последующим выполнением аффинного преобразования

$$b = M(a \cdot \gamma)^{-1} + \beta.$$

Здесь уже $a = \{a_0, a_1, a_2, a_3\}$ и $b = \{b_0, b_1, b_2, b_3\}$ – четырехбитные векторы (полубайты матрицы состояний), M – квадратная невырожденная матрица 4×4:

$$M = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix},$$

β – 4-битный вектор ($\beta \in \mathrm{GF}(2^4)$):
$\beta'' = (1 \quad 0 \quad 1 \quad 0)$.

Каждое значение выхода подстановки $b = \{b_0, b_1, b_2, b_3\}$ зависит как от входного состояния $a = \{a_0, a_1, a_2, a_3\}$, так и от случайного вектора $\gamma = \{\gamma_0, \gamma_1, \gamma_2, \gamma_3\}$, который задается значением циклового ключа. В результате осуществляется криптографическое преобразование данных, при котором происходит динамическое изменение S блоков (блоков нелинейных замен), с сохранением показателей их нелинейности.

Так, например, при $\gamma_i = 6 \rightarrow 0110$ функция нелинейного преобразования задается следующей табл. 2.

При $\gamma_i = k_0 = 0 \rightarrow 0000$ параметр γ_i принимается равным k_1. Если полубайт k_1 также равен нулю (0000), то $\gamma_i = 3 \rightarrow 0011$.

Таблица 1

Представление 4-битных массивов в двоичном и шестнадцатеричном виде

Дв	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Ш	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f

Таблица 2

Таблица подстановок, реализующая S-блок Baby-ADE при $\gamma_i = 6$

a	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
s(*a*)	a	2	0	6	f	1	c	4	E	b	7	d	9	5	3	8

Заметим, что при рассмотренном методе построения блоков нелинейных замен можно получить S-блоки с фиксированными точками (тождественными переходами) и противоположными точками (переходами в противоположные значения). Это недопустимо (нежелательно) для статических S-блоков, но может показаться, что если S-блоки зависят от секретного параметра, то наличие фиксированных точек существенно не повлияет на криптографическую стойкость шифра ввиду того, что криптоаналитик не знает моментов появления фиксированных точек. Однако сравнительно небольшое число возможных S-блоков (24 – 1 = 15 — число различных значений вектора параметров γ), а для большого шифра их будет 256, позволяет вычислить априорные вероятности существования фиксированных точек, что может способствовать построению атак на шифр. Отмеченный момент свидетельствует о необходимости тщательного подбора параметров аффинного преобразования.

Остается отметить, что в еще одном предложении по построению уменьшенной модели шифра Rijndael [3], найденной в открытой печати, используется S-блок, построенный по правилам, предложенным разработчиками Дайменом и Риджменом, что соответствует S-блоку шифра ADE при значении параметра γ = 1.

2.3 ShiftRow, π

В шифре mini-AES (как и в AES-е) операция π меняет входы во второй строке состояния, первая строка остается неизменной. Данное преобразование формально записывается так:

$$\begin{bmatrix} b_0 & b_2 \\ b_1 & b_3 \end{bmatrix} \xrightarrow{\pi} \begin{bmatrix} b_0 & b_2 \\ b_3 & b_1 \end{bmatrix},$$

что иллюстрирует рис. 4:

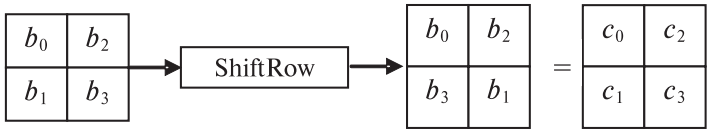

Рис. 4. Операция ShiftRow

Здесь уже a_i и b_i — элементы матриц-состояний (полубайты).

В шифре ADE в отличие от AES и эта операция строится ключезависимой. Для этого используются значения битов четности байта ключа k_2, с помощью которых осуществляется выбор строки матрицы состояния, в которой выполняется изменение положения байтов. Аналогичная процедура шифра Baby-ADE для выбора строки состояния, в которой меняются входы, также использует значение четности полубайта k_2 циклового ключа: если бит четности равен единице, то меняются местами полубайты 1-ой строки матрицы состояния, а если бит четности равен нулю, то меняются местами полубайты второй строки. Так, при значении бита четности $p = 0$ преобразование будет иметь вид: $\begin{bmatrix} b_0 & b_2 \\ b_1 & b_3 \end{bmatrix} \xrightarrow{\pi} \begin{bmatrix} b_0 & b_2 \\ b_3 & b_1 \end{bmatrix}$.

В данном случае оно тождественно соответствующему преобразованию шифра mini-AES. При значении функции четности $p = 1$ преобразование будет иметь вид: $\begin{bmatrix} b_0 & b_2 \\ b_1 & b_3 \end{bmatrix} \xrightarrow{\pi} \begin{bmatrix} b_2 & b_0 \\ b_1 & b_3 \end{bmatrix}$.

2.4 MixColumn, θ

Повторяя принципы ее реализации в AES-е, эта процедура в шифре mini-AES принимает каждый столбец матрицы состояния в качестве входа и умножает его на константную матрицу для получения нового выходного столбца, как показано на рис. 5. На этом рисунке c_0, c_1, c_2, c_3 и d_0, d_1, d_2, d_3 — элементы матриц-состояний входа и выхода соответственно рассматриваемого преобразования.

$$\begin{bmatrix} d_0 \\ d_1 \end{bmatrix} = \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix} \times \begin{bmatrix} c_0 \\ c_1 \end{bmatrix}, \quad \begin{bmatrix} d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix} \times \begin{bmatrix} c_2 \\ c_3 \end{bmatrix}.$$

Рис. 5. Операция MixColumn

В AES матрица преобразования строится на основе кода Рида-Соломона (8,4,5) над полем $\mathbf{GF}(2^8)$. В baby-ADE эквивалентом этому преобразованию использован расширенный $(16, 2, 15)$ код Рида-Соломона над полем $\mathbf{GF}(2^4)$. По определению это МДР-код, который имеет при заданных параметрах максимальное кодовое расстояние. Порождающая матрица кода в данном случае задается в виде $G = \begin{pmatrix} 0 & 1 & 2 & \dots & F \\ 1 & 1 & 2^2 & \dots & F^2 \end{pmatrix}$, где возведение в степень производится в поле $\mathbf{GF}(2^4)$, то есть элементы матрицы — полубайты.

Используя полубайт $k_1 = s$ циклового ключа (элемент поля $\mathbf{GF}(2^4)$), формируется невырожденная матрица $t = \begin{pmatrix} s & s^2 \\ s^2 & s^4 \end{pmatrix}$, которая является подматрицей определенной выше порождающей матрицы G расширенного $(16, 2, 15)$ кода Рида-Соломона. Свойство невырожденности (обратимости) для матрицы t обеспечивается ее размерностью, которая не превышает порядка элемента s в поле $\mathbf{GF}(2^4)$ [11]. В этом случае все столбцы и строки матрицы t получаются различными, и любая подматрица становится обратимой. Следует отметить, что данное свойство справедливо только в случае, если s не равно 0 или 1 (элементы $\mathbf{GF}(2^4)$ в 16-ричном представлении). Поэтому, в случае $k_1 = 0 \rightarrow 0000$ или $k_1 = 1 \rightarrow 0001$ в качестве элемента s принимается значение k_2; если $k_2 = 0$ или $k_2 = 1$, то s принимается равным 2 (элемент $\mathbf{GF}(2^4)$ в 16-ричном представлении).

Так, при $k_1 = 3 \rightarrow 0011$ матрица рассеивания примет вид: $t = \begin{pmatrix} 3 & 5 \\ 5 & 2 \end{pmatrix}$. Результатом выполнения

операции MixColumn является матричное умножение состояния справа на матрицу t над $\mathbf{GF}(2^4)$: $d = t \cdot c$.

2.5 KeyAddition, σ_{Ki}

В самом начале процедуры зашифрования и в конце каждого цикла состояние побитно складывается (т.е. по модулю 2) с цикловым ключом. Эти процедуры в шифрах Mini-AES и Baby-ADE полностью совпадают. Каждый бит входного блока $d = (d_0, d_1, d_2, d_3)$ складывается по модулю 2 с соответствующим битом циклового ключа $k_i = (k_0, k_1, k_2, k_3)$, для получения выходного блока $e = (e_0, e_1, e_2, e_3)$. Соответствующее преобразование иллюстрирует рисунок 6. Цикловой ключ формируется из секретного ключа K с помощью процедуры расширения (разворачивания) ключа, которая описана в следующем разделе. Для каждого бита операция XOR (сложение по модулю 2) имеет выходом 1, если соответствующие биты входного и выходного блока различны. Если биты входного и выходного блока совпадают, то выходной бит равен 0.

d_0	d_2		k_0	k_2		e_0	e_2
d_1	d_3	$\oplus$	k_1	k_3	$=$	e_1	e_3

Рис. 6. Операция KeyAddition

2.6 KeySchedule

Процедура расширения ключа для шифра Baby-ADE определена аналогично процедуре **KeySchedule** шифра Mini-AES: столбцы цикловых ключей вычисляются рекурсивно следующим образом:

$$w_0 = \begin{pmatrix} k_0 \\ k_1 \end{pmatrix} \quad w_1 = \begin{pmatrix} k_2 \\ k_3 \end{pmatrix}$$

$$w_{2i} = w_{2i-2} \oplus SubHalfBytes(reverse(w_{2i-1})) \oplus r_i$$

$$w_{2i+1} = w_{2i-1} \oplus w_{2i}$$

для $i = 1, 2, 3, 4$. Постоянные $r_i = \begin{pmatrix} 2^{i-1} \\ 0 \end{pmatrix}$ и функция **reverse()** заменяют два входа в столбец. Функция SubHalfBytes описана выше в подразделе 3.2. Заметим, что все сложения выполняются побитно по модулю 2. Наконец, для $i = 1, 2, 3, 4$ цикловой ключ K_i есть матрица, столбцами которой являются w_{2i} и w_{2i+1}. Кроме того, при расширении ключа рассчитываются зависимые от ключа параметры, которые используются в перечисленных выше процедурах. Так, в процедуре расширения ключа рассчитываются для каждого цикла изменяемые матрицы рассеивания, таблицы подстановок и функция четности, используемая как параметр при сдвиге строк состояния в процедуре ShiftRow. Для реализации динамического управления в блоках SubHalfBytes, MixColumn и ShiftRow берутся три полубайта ключа каждого цикла k_0, k_1 и k_2 соответственно.

2.7 Инверсный шифр

Инверсный шифр довольно легко может быть дедуцирован из информации, данной выше. Заметим здесь, что в инверсном шифре используются те же цикловые ключи, что и для шифрования, но в обратном порядке. Таблица подстановки инверсного шифра строится на основе таблицы прямого шифра. Пример подстановки инверсного шифра при $k_0 = \gamma_i = 6$ приведен в табл. 3.

Матрицы рассеивания t^{-1} для инверсного шифра рассчитываются при разворачивании ключей инвертированием матриц t. Так, для значения $k_1 = 3$ для матрицы рассеивания $t = \begin{pmatrix} 3 & 5 \\ 5 & 2 \end{pmatrix}$ инверсная матрица принимает вид $t^{-1} = \begin{pmatrix} 9 & C \\ C & 4 \end{pmatrix}$. При выполнении операции ShiftRow^{-1} следует помнить, что порядок ее выполнения определяется четностью 16-ричного числа – полубайта k_2 циклового ключа.

2.8 Примеры вычислений

Расширение ключа **6b5d** выполняется аналогично тому, как это осуществляется в шифре mini-AES:

$$w_0 = \begin{pmatrix} 6 \\ b \end{pmatrix}, \quad w_1 = \begin{pmatrix} 5 \\ d \end{pmatrix}$$

$$w_1 \xrightarrow{reverse} \begin{pmatrix} 5 \\ d \end{pmatrix} \xrightarrow{\gamma} \begin{pmatrix} 1 \\ e \end{pmatrix} \oplus w_0 = \begin{pmatrix} 7 \\ 5 \end{pmatrix} \oplus r_1 = \begin{pmatrix} 6 \\ 5 \end{pmatrix} = w_2$$

$$w_1 \oplus w_2 = \begin{pmatrix} 3 \\ 8 \end{pmatrix} = w_3$$

$$w_3 \xrightarrow{reverse} \begin{pmatrix} 8 \\ 3 \end{pmatrix} \xrightarrow{\gamma} \begin{pmatrix} 5 \\ b \end{pmatrix} \oplus w_2 = \begin{pmatrix} 3 \\ e \end{pmatrix} \oplus r_2 = \begin{pmatrix} 1 \\ e \end{pmatrix} = w_4$$

$$w_3 \oplus w_4 = \begin{pmatrix} 2 \\ 6 \end{pmatrix} = w_5$$

$$w_5 \xrightarrow{reverse} \begin{pmatrix} 6 \\ 2 \end{pmatrix} \xrightarrow{\gamma} \begin{pmatrix} 2 \\ 3 \end{pmatrix} \oplus w_4 = \begin{pmatrix} 3 \\ d \end{pmatrix} \oplus r_3 = \begin{pmatrix} 7 \\ d \end{pmatrix} = w_6$$

$$w_5 \oplus w_6 = \begin{pmatrix} 5 \\ b \end{pmatrix} = w_7$$

$$w_7 \xrightarrow{reverse} \begin{pmatrix} b \\ 5 \end{pmatrix} \xrightarrow{\gamma} \begin{pmatrix} f \\ e \end{pmatrix} \oplus w_6 = \begin{pmatrix} 8 \\ 3 \end{pmatrix} \oplus r_4 = \begin{pmatrix} 0 \\ 3 \end{pmatrix} = w_8$$

$$w_7 \oplus w_8 = \begin{pmatrix} 5 \\ 8 \end{pmatrix} = w_9$$

Таблица 3

Таблица подстановки, реализующая S-блок инверсного к Baby-ADE шифру при $\gamma_i = 6$

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$S^{-1}(x)$	2	5	1	e	7	d	3	a	f	c	0	9	6	b	8	4

Рассчитанные матрицы рассеивания и блоки нелинейных замен для 3-х циклов Baby-ADE (в последнем цикле операция MixColumn не выполняется) имеют следующий вид:

$$t_1 = \begin{pmatrix} 5 & 2 \\ 2 & 4 \end{pmatrix},\ t_2 = \begin{pmatrix} e & b \\ b & 9 \end{pmatrix},\ t_2 = \begin{pmatrix} d & e \\ e & b \end{pmatrix}.$$

Подстановки, реализующие S-блоки Baby-ADE в зависимости от цикловых ключей, приведены в табл. 4.

Пример шифрования для ключа 6*b*5*d* и открытого текста *d*8*e*5.

цикл	Старт	s	σ	Умнож. на t	+ Цикловый ключ	
Вход	$\begin{bmatrix} d & e \\ 8 & 5 \end{bmatrix}$				$+\begin{bmatrix} 6 & 5 \\ b & d \end{bmatrix}$	=
1	$\begin{bmatrix} b & b \\ 3 & 8 \end{bmatrix}$	$\begin{bmatrix} d & d \\ 6 & e \end{bmatrix}$	$\begin{bmatrix} d & d \\ e & 6 \end{bmatrix}$	$\begin{bmatrix} 3 & 0 \\ 4 & 2 \end{bmatrix}$	$+\begin{bmatrix} 6 & 3 \\ 5 & 8 \end{bmatrix}$	=
2	$\begin{bmatrix} 5 & 3 \\ 1 & a \end{bmatrix}$	$\begin{bmatrix} e & b \\ 4 & 6 \end{bmatrix}$	$\begin{bmatrix} b & e \\ 4 & 6 \end{bmatrix}$	$\begin{bmatrix} 2 & 4 \\ b & b \end{bmatrix}$	$+\begin{bmatrix} 1 & 2 \\ e & 6 \end{bmatrix}$	=
3	$\begin{bmatrix} 3 & 6 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} 7 & 4 \\ 5 & e \end{bmatrix}$	$\begin{bmatrix} 7 & 4 \\ e & 5 \end{bmatrix}$	$\begin{bmatrix} e & 2 \\ 4 & c \end{bmatrix}$	$+\begin{bmatrix} 7 & 5 \\ d & b \end{bmatrix}$	=
4	$\begin{bmatrix} 9 & 7 \\ 9 & 7 \end{bmatrix}$	$\begin{bmatrix} 5 & 7 \\ 5 & 7 \end{bmatrix}$	$\begin{bmatrix} 5 & 7 \\ 7 & 5 \end{bmatrix}$		$+\begin{bmatrix} 0 & 5 \\ 3 & 8 \end{bmatrix}$	=
Выход	$\begin{bmatrix} 5 & e \\ 4 & d \end{bmatrix}$					

Таким образом, зашифрование "открытого текста" *d*8*e*5 с ключом 6*b*5*d* дает шифртекст 54*ed*.

Для расшифрования шифртекста инверсным шифром в процедуре расширения ключа рассчитываются таблицы подстановок, инверсные к соответствующим таблицам прямого шифра. Они применяются в обратной последовательности, то есть подстановка, инверсная к таблице замен первого цикла, применяется в четвертом цикле инверсного шифра.

Подобным же образом рассчитываются и применяются матрицы рассеивания для трех циклов инверсного Baby-ADE (в последнем цикле операция MixColumn не выполняется). Также как и таблицы замен инверсного шифра, они рассчитываются из матриц прямого шифра и применяются в обратной последовательности: $t_3^{-1} = \begin{pmatrix} a & d \\ d & 1 \end{pmatrix}$ – в первом раунде, $t_2^{-1} = \begin{pmatrix} 8 & e \\ e & 1 \end{pmatrix}$ – во втором раунде и $t_1^{-1} = \begin{pmatrix} d & f \\ f & 3 \end{pmatrix}$ в третьем.

Пример расшифрования шифрограммы 54*ed* ключом 6*b*5*d*:

цикл	Старт	σ	$s^{-1}(x)$	+ Цикловый ключ		Умножение на t^{-1}	
Вход	$\begin{bmatrix} 5 & e \\ 4 & d \end{bmatrix}$			$+\begin{bmatrix} 0 & 5 \\ 3 & 8 \end{bmatrix}$	$\begin{bmatrix} 5 & 7 \\ 7 & 5 \end{bmatrix}$		=
1	$\begin{bmatrix} 5 & 7 \\ 7 & 5 \end{bmatrix}$	$\begin{bmatrix} 5 & 7 \\ 5 & 7 \end{bmatrix}$	$\begin{bmatrix} 9 & 7 \\ 9 & 7 \end{bmatrix}$	$+\begin{bmatrix} 7 & 5 \\ d & b \end{bmatrix}$	$\begin{bmatrix} e & 2 \\ 4 & c \end{bmatrix}$	$\begin{bmatrix} 7 & 4 \\ e & 5 \end{bmatrix}$	=
2	$\begin{bmatrix} 7 & 4 \\ e & 5 \end{bmatrix}$	$\begin{bmatrix} 7 & 4 \\ 5 & e \end{bmatrix}$	$\begin{bmatrix} 3 & 6 \\ 5 & d \end{bmatrix}$	$+\begin{bmatrix} 1 & 2 \\ e & 6 \end{bmatrix}$	$\begin{bmatrix} 2 & 4 \\ b & b \end{bmatrix}$	$\begin{bmatrix} b & e \\ 4 & 6 \end{bmatrix}$	=
3	$\begin{bmatrix} b & e \\ 4 & 6 \end{bmatrix}$	$\begin{bmatrix} e & b \\ 4 & 6 \end{bmatrix}$	$\begin{bmatrix} 5 & 3 \\ 1 & a \end{bmatrix}$	$+\begin{bmatrix} 6 & 3 \\ 5 & 8 \end{bmatrix}$	$\begin{bmatrix} 3 & 0 \\ 4 & 2 \end{bmatrix}$	$\begin{bmatrix} d & d \\ e & 6 \end{bmatrix}$	=
4	$\begin{bmatrix} d & d \\ e & 6 \end{bmatrix}$	$\begin{bmatrix} d & d \\ 6 & e \end{bmatrix}$	$\begin{bmatrix} b & b \\ 3 & 8 \end{bmatrix}$	$+\begin{bmatrix} 6 & 5 \\ b & d \end{bmatrix}$	$\begin{bmatrix} d & e \\ 8 & 5 \end{bmatrix}$		=
Выход	$\begin{bmatrix} d & e \\ 8 & 5 \end{bmatrix}$						

Таким образом, расшифрование шифртекста 54*ed* ключом 6*b*5*d* дает открытый текст *d*8*e*5. В следующем разделе мы предлагаем результаты реализации атаки невозможных дифференциалов на уменьшенные модели шифров mini-AES и baby-ADE.

3. ВЫПОЛНЕНИЕ АТАКИ НЕВОЗМОЖНЫХ ДИФФЕРЕНЦИАЛОВ С ИСПОЛЬЗОВАНИЕМ УМЕНЬШЕННЫХ МОДЕЛЕЙ ШИФРОВ MINI-AES И BABY-ADE

Представляется сначала уместным кратко напомнить сущность атаки с использованием невозможных дифференциалов. Более детальное ее описание можно найти в работах [16, 17].

Таблица 4

Таблицы подстановок для 4-х циклов Baby-ADE

x		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$s(x)$	1-й цикл ($\gamma_1 = 6$)	a	2	0	6	f	1	c	4	e	b	7	d	9	5	3	8
	2-й цикл ($\gamma_2 = 1$)	a	4	3	b	8	e	2	c	5	7	6	f	0	1	9	d
	3-й цикл ($\gamma_3 = 7$)	a	c	9	7	d	5	4	2	1	6	b	8	3	e	0	f
	4-й цикл ($\gamma_4 = 3$)	a	b	2	e	0	d	6	7	f	5	1	9	c	8	4	3

Таблица 5

Подстановки для 4 циклов инверсного Baby-ADE

x		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$s^{-1}(x)$	1-й цикл ($\gamma_1 = 3$)	4	a	2	f	e	9	6	7	d	b	0	1	c	5	3	8
	2-й цикл ($\gamma_2 = 7$)	e	8	7	c	6	5	9	3	b	2	0	a	1	4	d	f
	3-й цикл ($\gamma_3 = 1$)	c	d	6	2	1	8	a	9	4	e	0	3	7	f	5	b
	4-й цикл ($\gamma_4 = 6$)	2	5	1	e	7	d	3	a	f	c	0	9	6	b	8	4

Криптоанализ на основе невозможных дифференциалов был введен Бихамом и др. и сегодня рассматривается как один из эффективных методов построения атак на шифры. Эти атаки используют дифференциалы с вероятностью равной нулю, названные невозможными дифференциалами.

В обычной атаке дифференциального криптоанализа (DC атаке) с целью отличия шифра от случайной перестановки находятся характеристики высоковероятных дифференциалов для осуществления, в конечном счете, атаки с восстановлением ключа. В таких атаках биты подключа угадываются в циклах, окружающих дифференциал. Подключи, которые после частичного шифрования/расшифрования являются наиболее (или иногда, наименее) часто угадываемыми с помощью дифференциалов, принимаются как правильные. Новый метод, описанный Бихамом и др., наоборот, выполняет поиск дифференциалов, которые принимают никогда не случающиеся значения, а именно, значения, вероятности которых равны нулю. В варианте атаки с восстановлением ключа, если значение кандидата подключа приводит к выводу, что текстовая пара удовлетворяет такому дифференциалу, то подключ конечно будет неправильным. Эта процедура, называемая отсевом (*sieving*), находит правильный подключ путем исключения всех других подключей, которые удовлетворяют дифференциалу.

Один из методов построения атаки невыполнимых дифференциалов называется «*несовпадение посредине*» (*miss-in-the-middle*). Он состоит в комбинировании двух дифференциалов, каждый из которых выполняется с вероятностью 1, которые не могут достигаться одновременно. Поэтому комбинация этих дифференциалов приводит к противоречию. Как только такая комбинация обнаружена, она может использоваться как распознаватель для отбрасывания неправильных подключей и нахождения правильных подключей методом исключения. Другой подход, также предложенный Бихамом и др. в [17], базируется на полной структуре шифра. Процедура нахождения невыполнимых дифференциалов состоит в шифровании многих пар открытого текста, на всех возможных ключах, и отбрасывании всех получаемых разностей, потому что они достижимы. В результате путем исключения (опять отсева) остаются только дифференциалы, которые никогда не появляются. Однако такой поиск обычных невыполнимых дифференциалов требует перебора слишком многих дифференциалов и ключей. Для решения этой проблемы было предложено выполнять анализ уменьшенных вариантов шифров, в которых сохраняется характер их компонентов и полные структуры, но размер шифруемых блоков уменьшается до возможности осуществления исчерпывающего поиска невыполнимых дифференциалов.

Сохранение характера компонентов шифра означает, например, что большие перестановки заменяются малыми, большие функции заменяются меньшими, сохраняется также сходство линейных преобразований и других операций. Этот метод получил название техники уменьшения (*Shrinking technique*). Опираясь на сходство в структуре оригинального шифра и его уменьшенной модели, делается вывод о наличии для шифра невозможных дифференциалов, если таковые найдены для мини-версии данного шифра. Если атакующие не смогут найти невозможные дифференциалы для уменьшенного шифра, то и большой шифр не может быть атакован с использованием невозможных дифференциалов.

В настоящее время наиболее эффективным способом конструирования невозможного события, о котором шла речь выше, является сочетание методики (техники) уменьшения и техники встречи посередине, предложенное Бихамом. Эта методика предполагает наличие двух событий, каждое из которых всегда происходит, но вместе эти два события несовместны. Соединение этих событий, таким образом, что они приводят к противоречию посередине шифра, ведет к невозможному событию.

Базовая стратегия применения техники «встречи посередине» для нахождения невозможных дифференциалов описана Казумаро Аоки и Масаюки Канда в работе [16]. Алгоритм, реализующий эту стратегию, состоит из шести шагов:

Шаг 1. Выбрать входную разность X.

Шаг 2. Получить для этой входной разности все возможные выходные разности для r-того цикла Z_r.

Шаг 3. Найти множество позиций битовых разностей Z_r, чьи значения не меняются (всегда равны нулю или не равны нулю). Если такое множество не найдено, возвратиться к шагу 1. Если таких позиций не может быть найдено для всех входных разностей, то для данного шифра невозможные дифференциалы не существуют.

Шаг 4. Выбрать выходную разность Y шифра.

Шаг 5. Получить все возможные разности Z'_r для того же r-того цикла для выходной разности Y.

Шаг 6. Проверить, всегда ли являются ненулевыми (нулевыми) разности для тех же позиций, которые получены на шаге 3. Если это так, то это означает, что невозможные дифференциалы относительно входной разности X и выходной разности Y найдены. В противоположном случае возвратиться к шагу 4, и так действовать, пока не будут проверены все выходные разности. Если проверка не проведена для всех входных разностей, возвратиться к шагу 1.

Далее будут представлены результаты применения этой методики для нахождения невозможных дифференциалов шифра baby-ADE.

3.1 Атака на 5-цикловый mini-AES

Шифр мини-АЕС имеет только два цикла, что делает атаку на него тривиальной, поэтому мы рассмотрим шифр с большим количеством циклов. При конструировании невозможного события

стремятся покрыть как можно больше циклов. Однако, забегая вперед, можно отметить, что, как и в оригинальном шифре AEC, 4 цикла и в этом случае оказываются тем максимумом, который удается достигнуть и для шифра мини-AEC.

При построении атаки невозможных дифференциалов на пяти цикловой мини-AEC мы здесь воспользуемся идеями построения атаки на полную структуру шифра AEC, рассмотренную в работе [17]. Невозможные дифференциалы при развиваемом в этой работе подходе удается найти без применения переборного алгоритма, представленного ранее. Здесь автор использует свойства преобразований шифра AEC, установленные в ряде работ [18 и др.], и мы сначала просто проиллюстрируем развиваемый в работе [17] подход применительно к построению атаки невозможных дифференциалов на мини-AEC.

Следуя идеям работы [17], рассмотрим преобразования 16-битных блоков данных (открытых текстов), осуществляемые 4-цикловым mini-AES-ом. Можно убедиться, что если пара открытых текстов отличается только одним полубайтом, то шифртексты не могут быть равными (совпадающими) в одной из двух комбинаций полубайт: (1, 4) или (2, 3). Это обусловлено тем, что разность до первого MixColumn в этом случае поддерживается только в одном полубайте, и, следовательно, разность после преобразования MixColumn переходит в отличия одного из столбцов, а *после второго MixColumn состояние имеет отличия во всех полубайтах*. Отмеченное свойство иллюстрирует рис. 7, на котором изображены промежуточные состояния при зашифровании пары блоков открытых текстов, отличающихся первым полубайтом – (2, 3, 6, 1) и (a, 3, 6, 1).

С другой стороны, если блоки шифртекстов равны в одной из 2 невозможных комбинаций, то после 3-го MixColumn данные равны в одном столбце. Поэтому *после второго MixColumn существуют* 2 *полубайта, которые равны друг другу*. На рис. 8 приведены промежуточные состояния, полученные при зашифровании блоков открытого текста, результирующегося в блоки шифртекста, отличающиеся в первом и последнем полубайтах.

Это и есть противоречие, поскольку ранее мы показали, что после второго MixColumn состояние имеет отличия во всех полубайтах. Эти два события (свойства), конечно же, несовместимы.

Атака на mini-AES расширенный до 5 циклов состоит в исключении неправильных ключей $k_0k_1k_2k_3$, применяемых в дополнительном преобразовании, выполняемом перед первым циклом 4-х циклового mini-AES, рассмотренным выше. Ключи являются неправильными, если для них выполняется невозможное свойство для последних 4 циклов. За счет расширения mini-AES дополнительным циклом (в начале шифра) входному блоку открытого текста для последующих 4 циклов mini-AES будет соответствовать результат выполнения MixColumn первого цикла, полученного 5-циклового mini-AES. При проведении атаки определение ключа шифрования проводится в 2 приема: сначала определяются полубайты ключа k_0k_3, а затем полубайты k_1k_2. Последовательность применения этих этапов не играет роли.

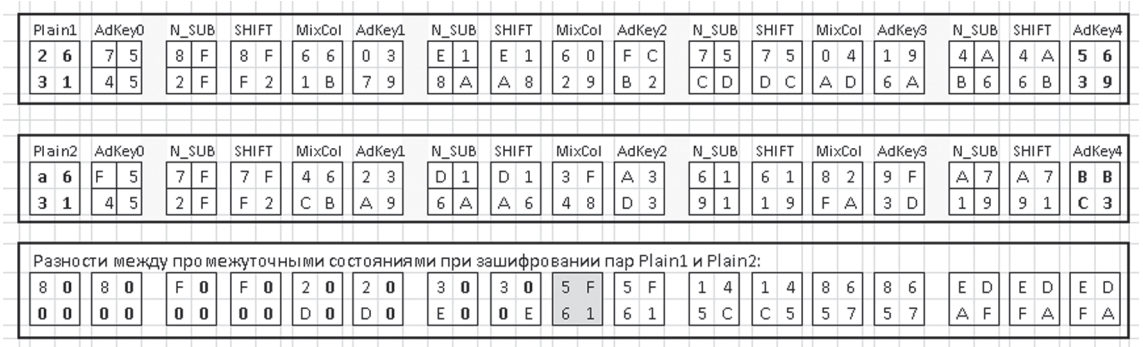

Рис. 7. Промежуточные состояния при зашифровании пары открытых текстов и разности между ними

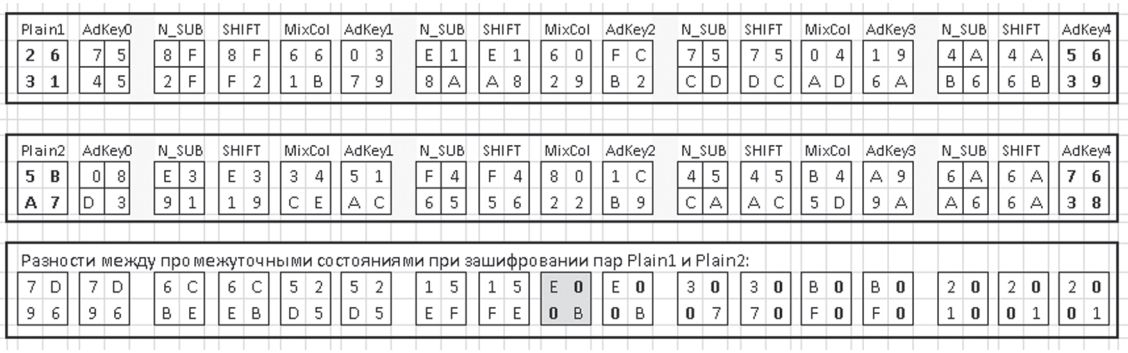

Рис. 8. Промежуточные состояния при зашифровании пары открытых текстов и разности между ними при условии равенства полубайт в позициях (2,3)

Рассмотрим, например, порядок определения полубайтов ключа k_0k_3 (для 5-цикловой версию шифра mini-AES). На стадии предвычислений рассматриваются все пары полубайтов (a, b), (a', b), $a \neq a'$ первого столбца состояния после применения операции MixColumn первого цикла. Выясняется, какие из входных разностей пар открытых текстов приводят к таким значениям разностей после 1-го цикла, которые ведут к выполнению первого из условий невозможного события — и одна из разностей после MixColumn третьего цикла не равна нулю. Для этого проводится «расшифрование» пар полученных после первого цикла, т.е. выполняются операции MixColumn^{-1}, ShiftRow^{-1}, SubHalfBytes^{-1} и создается хеш-таблица, состоящая из одного из входов x в SubHalfBytes и результата операции XOR пары входов $x \oplus y$, где x, y — пара входов в SubHalfBytes. Существует 2^8 возможных значений $x \oplus y$, что дает $\approx 2^{12}$ значений в таблице, так что в среднем получается $\approx 2^4$ значений x, которые соответствуют каждому $x \oplus y$.

При осуществлении атаки рассматриваются 2^8 выбранных открытых текстов, которые предполагают все возможные значения пары полубайтов (p_0p_3) и 0 в остальных полубайтах. Для 2^8 выбранных открытых текстов получается около $(2^8)^2/2 = 2^{15}$ пар, так что имеется порядка $2^{15}/2^2 = 2^{13}$ пар, в которых шифртекст соответствует одной из "невозможных комбинаций". Для этих пар входов в SubHalfBytes рассчитывается побитовая сумма $x \oplus y$, и с использованием хеш-таблицы выявляется около 2^4 возможных значений x, которые соответствуют рассчитанным значениям $x \oplus y$. Этот процесс позволяет обнаружить около 2^4 возможных неверных значений применением операции XOR к открытому тексту и x. Вследствие коллизий количество оставшихся неверных ключей составит $2^8(1-2^{-3})^{2^6} \approx 2^8(e^{-1})^{2^3} = 2^8e^{-8} \approx 0$, так что останутся только верные ключи. Таким же образом можно получить остальные полубайты ключа — k_1k_2. В цитируемой работе отмечается, что оставшееся количество неправильных пар очень мало и поэтому может быть использовано меньшее количество выбранных текстов.

Сложность данной атаки. На стадии предвычислений выполняется цикл расшифрования одного столбца для 2^{12} пар. Это эквивалентно $\frac{1}{2 \cdot 5} \cdot 2^{13} \approx 2^{10}$ зашифрованиям. Полученная в результате предвычислений хеш-таблица требует для запоминания около 2^7 байт памяти.

Атака требует 2^8 выбранных блоков открытого текста, которые составят 2^{15} пар. Для каждой пары просматривается таблица и получается около 2^4 значений x, из которых рассчитываются соответствующие значения ключа. Эти значения затем удаляются из таблицы возможных ключей. Временная сложность этой стадии (и атаки) около 2^8 зашифрований. В итоге, атака требует 2^8 выбранных блоков открытого текста, 2^8 зашифрований, 2^7 байт памяти (включая 2^8 бит для таблицы удаленных ключей) и 2^{10} предвычислений.

3.2 Атака на 5-цикловый baby-ADE

Рассмотрим особенности проведения этой атаки на 5-цикловую версию шифра baby-ADE. Вследствие того, что структура цикловых преобразований одинакова, то есть цикловые функции состоят из блоков нелинейных замен той же размерности, а также линейных блоков ShiftRow и MixColumn, которые обеспечивают рассеивание подобное шифру mini-AES, 4-цикловые невозможные дифференциалы существуют и для шифра baby-ADE. Это подтверждается экспериментально. В процессе экспериментов была реализована атака невозможных дифференциалов в соответствии с алгоритмом, приведенным в начале раздела, а также атака, использующая структуру цикловой функции. Обе атаки показали наличие невозможных дифференциалов для четырех циклов шифра и их отсутствие для большего числа циклов.

Следует отметить, что поскольку на стадии предвычислений используются инверсные процедуры для расшифрования первого цикла шифра, то поскольку в шифре baby-ADE эти процедуры параметризированы секретным ключом, очевидно, что в процесс предвычислений вносятся определенные усложнения. Имеется в виду, что необходимо производить предвычисления всех вариантов возможных структур. По нашим подсчетам, их количество составляет $2(2^4-1)(2^4-2) = 420 \approx 2^{8,5}$. Следовательно, во столько же раз повышается временная сложность проведения атаки.

Экстраполируя результат на полный шифр ADE, с учетом увеличения (за счет динамики) числа подлежащих анализу дополнительных вариантов структур, для степени возрастания объема предвычислений можно получить оценку $4(2^8-1)(2^8-2) = 259080 \approx 2^{18}$.

Следует, также отметить, что в атаке «Квадрат», рассмотренной в работе [9], необходимым этапом также является проведение расшифрования 4-го цикла с использованием предполагаемого значением циклового ключа 4-го цикла. Использование динамически изменяемых блоков нелинейных замен и блоков линейного рассеивания в общем случае, по нашему мнению, также усложнит и эту атаку.

ВЫВОДЫ

Разработанная мини-версия алгоритма – кандидата на национальный стандарт шифрования ADE, – Baby-ADE, позволяет выполнить исследования криптографических свойств недоступных для полной версии ADE. В условиях масштабируемости больших шифров использование их малых копий является во многих случаях едва ли не единственной возможностью получения оценок стойкости больших шифров на основе сравнения показателей их уменьшенных версий. В частности, представленные результаты проведения атаки не-

возможных дифференциалов на шифры Baby-AES и Baby-AES подтвердили, что эти шифры имеют самое большее 5-цикловые невозможные дифференциалы и, следовательно, шифры Baby-AES и Baby-ADE неуязвимы атакам криптоанализа, использующим все известные алгоритмы распознавания невозможных дифференциалов.

Сравнительный анализ применения к мини-шифрам mini-AES и baby-ADE атаки с использованием невозможных дифференциалов и атаки «Квадрат» показывает, что введение зависимости внутренних операций цикловых преобразований от ключа позволяют значительно повысить временную сложность этих атак, что приведет к повышению запаса стойкости (security margin) шифров.

Литература

[1] Final report of European project number IST-1999-12324, named New European Schemes for Signatures, Integrity, and Encryption, April 19, 2004— Version 0.15 (beta), Springer-Verlag

[2] S. Landau, Polynomials in the Nation's Service: Using Algebra to Design the Advanced Encryption Standard, THE MATHEMATICAL ASSOCIATION OF AMERICA, 111 (February 2004), pp. 89–117.

[3] A Description of Baby Rijndael, ISU CprE/Math 533; NTU ST765-U, February 19, 2003.

[4] Шнайер Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. – Пер. с англ.: М.: Издательство ТРИУМФ, 2002 – 816 с.

[5] E. Biham, A. Shamir. Differential Cryptanalysis of DES-like Cryptosystems. Journal of Cryptology, Vol. 4 No.1, 1991, pp. 3-72.

[6] P. S. L. M. Barreto and V. Rijmen, "The Khazad legacy-level block cipher." Primitive submitted to NESSIE, Sept. 2000.

[7] X. Lai and J. L. Massey, "IDEA." Primitive submitted to NESSIE by R. Straub, MediaCrypt AG, Sept. 2000.

[8] J. Jonsson and B. S. Kaliski, Jr, "RC6 block cipher." Primitive submitted to NESSIE by RSA, Sept. 2000.

[9] Raphael Chung-Wei Phan, Mini Advanced Encryption Standard (Mini-AES): A Tested for Cryptanalysis Students, Cryptologia, XXVI (4), 2002. L. R. Knudsen, "The number of rounds in block ciphers." Public report, NESSIE, 2000.

[10] L. R. Knudsen, "Non-random properties of reduced-round Whirlpool." Public report, NESSIE, 2002.

[11] Блейхут Р.Теория и практика кодов, контролирующих ошибки – М.: Мир, 1986.

[12] Кузнецов А.А., Сергиенко Р.В., Наумко А.А. Симметричный криптографический алгоритм ADE (Algorithm of Dynamic Encryption) Радиотехника: Всеукр. межвед. науч.-техн. сб. – Харьков: ХТУРЭ. – 2008. (в печати).

[13] Nicolas T. Courtois. How Fast can be Algebraic Attacks on Block Ciphers? Cryptology ePrint Archive, Report 2006/168, 2006. available at: http://eprint.iacr.org/2006/168.pdf.

[14] S. Murphy and M. J. Robshaw. Comments on the Security of the AES and the XSL Technique, 2002.

[15] MuRo02Cr] S. Murphy and M. J. Robshaw. Essential Algebraic Structure within the AES. In Advances in Cryptology. CRYPTO 2002, volume 2442 of Lecture Notes in Com. puter

[16] Science, pages Springer Verlag, Heidelberg, 2002, pp. 1–16.

[17] Kazumaro Aoki and Masayuki Kanda. Search for Impossible Differential of E2.

[18] E. Biham and N. Keller, "Cryptanalysis of reduced variants of Rijndael." In Official public comment for Round 2 of the Advanced Encryption Standard development effort, 2000. Available at http://csrc.nist.gov/encryption/aes/round2/conf3/pa.

[19] N. Ferguson, R. Schroeppel, and D. Whiting. A Simple Algebraic Representation of Rijndael. In Selected Areas of Cryptography: 8th Annual International Workshop, volume 2259 of Lecture Notes in Computer Science, page pp. 103 ff. Springer Verlag Heidelberg, 2001.

Поступила в редколлегию 11.09.2008

Долгов Виктор Иванович, доктор технических наук, профессор кафедры «Безопасности информационных технологий» ХНУРЭ. Область научных интересов: математические методы защиты информации.

Кузнецов Александр Александрович, доктор технических наук, старший научный сотрудник. Область научных интересов: алгебраическая теория блоковых кодов, криптография и теория аутентификации, методы обеспечения помехоустойчивости, имитостойкости и скрытности каналов управления и связи.

Сергиенко Роман Викторович, соискатель ученой степени кандидата технических наук по специальности системы защиты информации. Область научных интересов: алгебраическая теория блоковых кодов, криптографические средства защиты информации.

Белоковаленко Андрей Леонидович, программист-разработчик ООО «ФулкрумВеб». Область научных интересов: теоретические основы криптографической защиты информации.