МЕТОДЫ, ПРОТОКОЛЫ И СРЕДСТВА КРИПТОГРАФИЧЕСКИХ ПРЕОБРАЗОВАНИЙ

УДК 681.3.06

И. Д. ГОРБЕНКО, д-р техн. наук, Д. С. БАЛАГУРА

ИССЛЕДОВАНИЕ СВОЙСТВ И ВЫБОР ПАРАМЕТРОВ СХЕМ ШИФРОВАНИЯ, РЕАЛИЗОВАННЫХ НА ОСНОВЕ ПРОТОКОЛОВ ДИФФИ-ХЕЛЛМАНА

В различных коммерческих и банковских технологиях значительное распространение получают криптографические преобразования, получившие название направленных шифров. Такие шифры используются для реализации состоятельных протоколов управления ключами, аутентификации, разделения секрета и др. Криптографические преобразования могут строиться на основе таких схем: схема направленного шифрования RSA, которая представлена на рис. 1 (на рис. 1: M – сообщение; O_O – открытый ключ получателя; $Д_П$ – личный ключ получателя), схема направленного шифрования Эль-Гамаля [1], комбинированные схемы и схемы направленного шифрования, базирующиеся на примитиве Диффи-Хеллмана [2]. К последним относятся схемы с преобразованиями в полях, кольцах и группе точек эллиптических кривых [3]. Наибольшее распространение получают схемы направленного шифрования на основе примитива Диффи-Хеллмана с преобразованиями в группе точек эллиптических кривых. Это объясняется меньшей сложностью таких преобразований и, как следствие, более высокой скоростью при сохранении стойкости. Сохраняя же скорость преобразований, можно существенно повысить стойкость. Однако такие схемы являются схемами направленного шифрования с оговоркой, так как для выполнения операций зашифрования и расшифрования используются гамма-последовательности на базе общих секретов. Общие секреты могут формироваться на долговременных и/или сеансовых ключах. Общая схема реализации шифрования на основе примитивов Диффи-Хеллмана представлена на рис. 2 (на рис. 2: KDF – алгоритм выработки гаммы шифрующей; *д_д* – дополнительные данные; Q_O – открытый ключ отправителя; $Q_П$ – открытый ключ получателя; D_O – личный ключ отправителя; $D_П$ – личный ключ получателя).

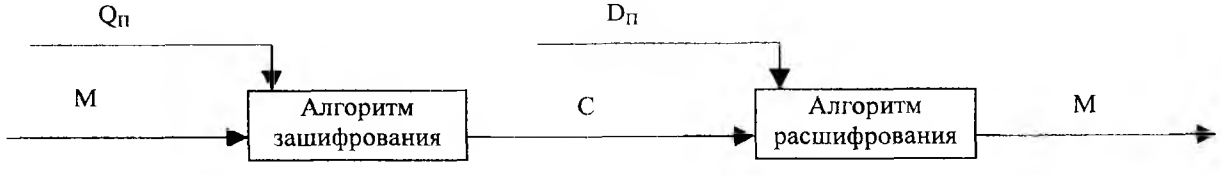

Рис. 1

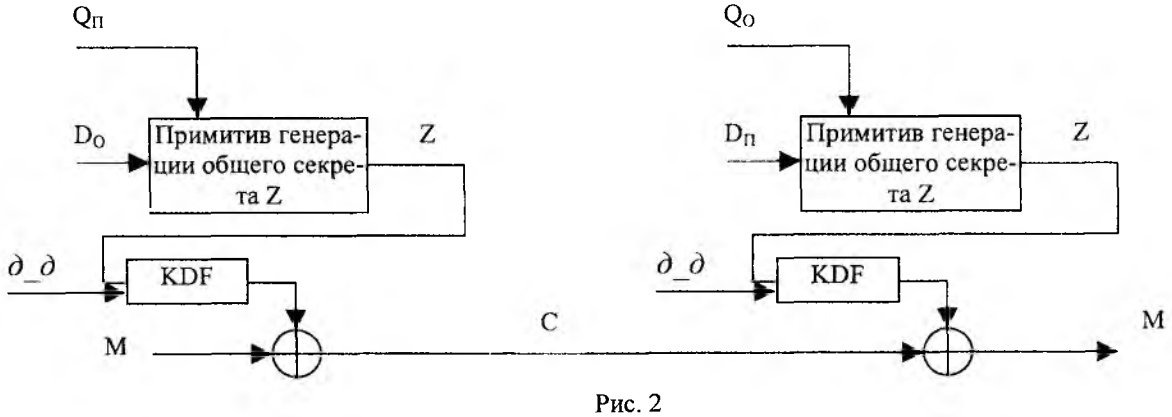

Рис. 2

Проведенный анализ показал, что стойкость шифрования схемы, представленной на рис. 2, в определяющей мере зависит от свойства гаммы шифрующей, формируемой KDF. Однако в известных источниках нет данных обоснований выбора схем KDF, методик исследований свойств гамм шифрования и выбора наиболее предпочтительных схем.

Целью настоящей статьи является проведение теоретического обоснования и практических исследований схем направленного шифрования в группе точек эллиптических кривых на основе примитивов Диффи-Хеллмана, а также разработка рекомендаций по их практическому применению.

1 Методы построения алгоритмов формирования гамм шифрующих

К настоящему времени известно несколько методов построения функций формирования гамм шифрующих. Логика преобразований этих функций может базироваться на однонаправленных хэш-функциях (ГОСТ 34.311-95, SHA-1, SHA-2, MD-5), на симметричных блочных алгоритмах, работающих в режиме выработки гаммы шифрующей, а также на основе поточных шифров.

При выработке гаммы шифрующей с использованием симметричных алгоритмов используется соответствующий алгоритм блочного шифрования в режиме выработки гаммы. При этом вектор инициализации и ключ могут формироваться из общего секрета либо вектор инициализации может быть заранее определён или выработан абонентами. При выработке гаммы шифрующей с использованием поточных шифров общий секрет используется в качестве вектора инициализации.

При выработке гаммы шифрующей с использованием однонаправленных хэш-функций используют префиксно-суффиксный вариант подстановки ключей, причём в качестве ключей применяют общий секрет, а также при необходимости дополнительные данные. При использовании для генерации гаммы шифрующей хэш-функции SHA-1 выполняются следующие этапы:

1. Выполняются шаги 2 и 3 до тех пор, пока гамма не достигнет размера, который на 4 и менее 32-битных блоков будет меньше размера шифруемой информации.
2. Вычисляется Γ_i = SHA-1(Дополнительные данные||Общий секрет||i).
3. Увеличивается значение счётчика (i++).
4. Если размер гаммы меньше размера шифруемой информации, выполнить генерацию неполного блока по формуле аналогично шагу 2. Уменьшить последний блок до необходимой длины.

2 Оценка безопасности функций развёртывания гаммы шифрующей (KDF)

При использовании функций формирования гаммы шифрующей на основе хэш-функций основными атаками на системы направленного шифрования могут быть атаки, основанные на коллизиях. Следовательно, для определения опасности таких атак необходимо дать оценку вероятностей возникновения коллизии. Как было показано в [4], вероятность возникновения коллизии подчиняется «парадоксу» о дне рождения. Существуют два варианта возникновения коллизий: возникновение коллизии в одной выходной последовательности и возникновение коллизии в разных выходных последовательностях. Дадим оценку вероятности появления коллизий в одной выходной последовательности.

Постановка задачи. Пусть имеется некоторая функция преобразования H

$$h=H(M), \qquad (1)$$

где M есть данные (информация) произвольной длины l_M, причем h может принимать $n=2^m$ значений независимо от длины l_M. Необходимо определить число k случайных сообщений M_i, которые необходимо подать на вход преобразователя H, чтобы с вероятностью $P_э$ состоялось хотя бы одно совпадение вида (1), т.е. состоялась коллизия.

Проведенный анализ показал [4], что задача 1 может быть решена с использованием «парадокса» о дне рождения, но при подробном рассмотрении выясняется, что она носит более общий характер. В нашем случае имеется выборка из k значений целочисленной случайной величины с равновероятным законом распределения, причем она может принимать значения от 1 до $n=2^m$, а $k \le n$. При этих условиях необходимо найти вероятность $P(n,k)$ того, что среди значений $H(M)$ выборки по крайней мере две совпадают, т.е.

$$H(M_i)=H(M_j).$$

Для решения задачи 1 найдем вероятность того, что в группе из k событий не состоится коллизия, т.е. соотношение (1) не выполнится ни разу. Обозначим эту вероятность как $R(n,k)$. Ясно, что $P(n,k)$ и $R(n,k)$ составляют полную группу событий, т.е.

$$P(n,k)+R(n,k)=1$$

и

$$P(n,k)=1-R(n,k). \tag{3}$$

Далее найдем общее число N различных способов, которыми можно получить k значений без повторений. Для первого элемента имеем n значений без повторений, для второго n-1, третьего – n-2 и т.д., для k-го (n-k+1). Поэтому общее число способов, при которых совпадений вида (1) нет, равно

$$N = n\cdot(n-1)(n-2)..(n-k+1) = \frac{n!}{(n-k)!}. \tag{4}$$

Проведя преобразования, аналогичные [4], мы получим общие формулы для определения количества сообщений, которые нужно подать на вход, чтобы с вероятностью P_k состоялась коллизия (5), а также вероятность осуществления коллизии при заданных значениях k и n (6).

$$k^2 = -2n\ln(1-P_k) \tag{5}$$

или

$$k = \sqrt{2\ln\left(\frac{1}{1-P_k}\right)\cdot n} = 1{,}41\sqrt{\ln\left(\frac{1}{1-P_k}\right)\cdot n}, \quad P_k = 1-e^{-\frac{k^2}{2n}}. \tag{6}$$

Здесь P_k – вероятность, с которой может возникнуть коллизия, а k – количество сформированных блоков гаммы.

Важной также является задача оценки вероятности появления коллизий в нескольких выходных последовательностях, т.е. появление одинаковых блоков гамм шифрующих длиной m битов [4].

Постановка задачи. Пусть на выходе преобразователя H из полного множества значений $n=2^m$ формируются k случайных значений функции преобразования (2), причем $k \le n$ и k подчиняются равновероятному закону распределения. Пусть выполнено z экспериментов, в каждом из которых получено k значений h. Обозначим реализации двух экспериментов соответственно как X и Y, причем $X=(x_1,x_2,\ldots,x_k)$ и $Y=(y_1,y_2,\ldots,y_k)$. Необходимо найти вероятность $P(n,k)$ того, что эти множества содержат в себе хотя бы по одному элементу x_i и y_j таких, что $x_i=y_j$.

Рассмотрим решение этой задачи, опираясь на результаты задачи 1. Вторая задача возникает при рассмотрении и выборе способов создания коллизий. Например, в нашей постановке эта задача имеет смысл, если рассмотрение сразу двух или большего числа множеств

из k реализаций позволяет ускорить процесс создания коллизий или извлечь полезную для криптоаналитика информацию для выполнения последующих атак.

В нашем случае событие $x_i=y_j$ может состояться с вероятностью $1/n$. Поэтому вероятность того, что $x_i \neq y_j$

$$Q(x_i \neq y_j) = 1 - \frac{1}{n}. \tag{21}$$

Если Y включает в себя k событий, то вероятность того, что все значения $y_1, y_2, \ldots, y_k$ не совпадут с x_i, может быть вычислена как

$$Q(x_i \neq Y) = \left(1 - \frac{1}{n}\right)^k. \tag{22}$$

Вероятность того, что хотя бы одно значение Y совпадет с x_i, есть

$$R(x_i \in Y) = 1 - \left(1 - \frac{1}{n}\right)^k. \tag{23}$$

Проведя необходимые преобразования, получим общую формулу расчёта количества элементов в последовательностях для получения совпадения с вероятностью P_k:

$$k = \sqrt{n \ln\left(\frac{1}{1 - P_k}\right)}. \tag{29}$$

Как видно из формулы, для совпадения с определённой вероятностью двух блоков из разных последовательностей необходимо сгенерировать последовательности в корень из двух раз короче, чем одиночная последовательность для задачи типа 1. То есть общее количество блоков при равной вероятности коллизии в задаче типа 1 меньше, чем в задаче типа 2.

В табл. 1 приведены оценки значений величины k, которые приводят к коллизиям с вероятностью коллизии P_k для случая использования в качестве KDF функций хэширования SHA-1, SHA-2(256), SHA-2(384), SHA-2(512), ГОСТ 34.311-95, ГОСТ 28.147-89 и Rijndael (длины ключей 128, 192 и 256 битов).

Таблица 1

P_k	10^{-16}	10^{-12}	10^{-6}	0,1	0,5	0,99	$1\text{-}10^{-5}$	$1\text{-}10^{-10}$
SHA-1	$1{,}9 \cdot 2^{53}$	$1{,}5 \cdot 2^{59}$	$1{,}47 \cdot 2^{70}$	$0{,}45 \cdot 2^{80}$	$1{,}18 \cdot 2^{80}$	$3{,}03 \cdot 2^{80}$	$4{,}78 \cdot 2^{80}$	$6{,}77 \cdot 2^{80}$
SHA-2 (256)	$1{,}9 \cdot 2^{101}$	$1{,}5 \cdot 2^{107}$	$1{,}47 \cdot 2^{118}$	$0{,}45 \cdot 2^{128}$	$1{,}18 \cdot 2^{128}$	$3{,}03 \cdot 2^{128}$	$4{,}78 \cdot 2^{128}$	$6{,}77 \cdot 2^{128}$
SHA-2 (384)	$1{,}9 \cdot 2^{165}$	$1{,}5 \cdot 2^{171}$	$1{,}47 \cdot 2^{182}$	$0{,}45 \cdot 2^{192}$	$1{,}18 \cdot 2^{192}$	$3{,}03 \cdot 2^{192}$	$4{,}78 \cdot 2^{192}$	$6{,}77 \cdot 2^{192}$
SHA-2 (512)	$1{,}9 \cdot 2^{229}$	$1{,}5 \cdot 2^{235}$	$1{,}47 \cdot 2^{246}$	$0{,}45 \cdot 2^{256}$	$1{,}18 \cdot 2^{256}$	$3{,}03 \cdot 2^{256}$	$4{,}78 \cdot 2^{256}$	$6{,}77 \cdot 2^{256}$
ГОСТ 34.311-95	$1{,}9 \cdot 2^{101}$	$1{,}5 \cdot 2^{107}$	$1{,}47 \cdot 2^{118}$	$0{,}45 \cdot 2^{128}$	$1{,}18 \cdot 2^{128}$	$3{,}03 \cdot 2^{128}$	$4{,}78 \cdot 2^{128}$	$6{,}77 \cdot 2^{128}$
ГОСТ 28.147-89	$1{,}9 \cdot 2^{5}$	$1{,}5 \cdot 2^{11}$	$1{,}47 \cdot 2^{22}$	$0{,}45 \cdot 2^{32}$	$1{,}18 \cdot 2^{32}$	$3{,}03 \cdot 2^{32}$	$4{,}78 \cdot 2^{32}$	$6{,}77 \cdot 2^{32}$

Продолжение табл. 1

P_k	10^{-16}	10^{-12}	10^{-6}	0,1	0,5	0,99	$1-10^{-5}$	$1-10^{-10}$
Rijndael (128 бит)	$1,9 \cdot 2^{37}$	$1,5 \cdot 2^{43}$	$1,47 \cdot 2^{54}$	$0,45 \cdot 2^{64}$	$1,18 \cdot 2^{64}$	$3,03 \cdot 2^{64}$	$4,78 \cdot 2^{64}$	$6,77 \cdot 2^{64}$
Rijndael (192 бит)	$1,9 \cdot 2^{69}$	$1,5 \cdot 2^{75}$	$1,47 \cdot 2^{86}$	$0,45 \cdot 2^{96}$	$1,18 \cdot 2^{96}$	$3,03 \cdot 2^{96}$	$4,78 \cdot 2^{96}$	$6,77 \cdot 2^{96}$
Rijndael (256 бит)	$1,9 \cdot 2^{101}$	$1,5 \cdot 2^{107}$	$1,47 \cdot 2^{118}$	$0,45 \cdot 2^{128}$	$1,18 \cdot 2^{128}$	$3,03 \cdot 2^{128}$	$4,78 \cdot 2^{128}$	$6,77 \cdot 2^{128}$

В табл. 2-8 приведены оценки возникновения вероятностей появления коллизий в зависимости от числа k, т.е. числа блоков гаммы шифрующей, при которых наступает коллизия с вероятностью P_k. Оценки приведены для алгоритмов ГОСТ 28147-89; Rijndael с длиной блока 128бит; SHA-1; Rijndael с длиной блока 192бит; Rijndael 256бит, SHA-2 256бит, ГОСТ-34.11-95; SHA-2 384бит; SHA-2 512бит. Так, для ГОСТ 28147-89 коллизия наступает с $P_k \approx 1$ уже при $k \geq 10^{12}$, а для Rijndael(128) при $k \geq 10^{20}$. Таким образом, Rijndael по сравнению с ГОСТ 28147-89 имеет существенно большую стойкость против коллизий, а также существенно меньшую вероятность перекрытия шифра. Это объясняется большей длиной блока у Rijndael. При использовании для реализации KDF функции хэширования SHA-1 с длиной хэш-функции 160 битов реальные коллизии возможны при $k \geq 10^{23}$, для SHA-2 с длиной хэш-функции 256 битов при $k \geq 10^{37}$, при длине хэш-функции 384 бита при $k \geq 10^{56}$, при длине хэш-функции 512 бита при $k \geq 10^{75}$. Алгоритм шифрования ГОСТ 34.311-95 обеспечивает такую же степень защиты от коллизий, как и SHA-2 при длине хэш-функции 256 битов.

Таблица 2

K	10	10^2	10^3	10^4	10^5	10^6
P_k	$2,71 \cdot 10^{-18}$	$2,71 \cdot 10^{-16}$	$2,71 \cdot 10^{-14}$	$2,71 \cdot 10^{-12}$	$2,71 \cdot 10^{-10}$	$2,71 \cdot 10^{-8}$

K	10^7	10^8	10^9	10^{11}	10^{12}	10^{13} и более
P_k	$2,71 \cdot 10^{-6}$	$2,71 \cdot 10^{-4}$	$2,71 \cdot 10^{-2}$	0,933	≈ 1	≈ 1

Таблица 3

K	$10-10^9$	10^{10}	10^{11}	10^{12}	10^{13}	10^{14}
P_k	≈ 0	10^{-19}	$1,47 \cdot 10^{-17}$	$1,47 \cdot 10^{-15}$	$1,47 \cdot 10^{-13}$	$1,47 \cdot 10^{-11}$

K	10^{15}	10^{16}	10^{17}	10^{18}	10^{19}	10^{20} и более
P_k	$1,47 \cdot 10^{-9}$	$1,47 \cdot 10^{-7}$	$1,47 \cdot 10^{-5}$	$1,47 \cdot 10^{-3}$	$1,47 \cdot 10^{-1}$	≈ 1

Таблица 4

K	$10-10^{14}$	10^{15}	10^{16}	10^{17}	10^{18}	10^{19}
P_k	≈ 0	$3,25 \cdot 10^{-19}$	$3,42 \cdot 10^{-17}$	$3,42 \cdot 10^{-15}$	$3,42 \cdot 10^{-13}$	$3,42 \cdot 10^{-11}$

Таблица 4 – продолжение

K	10^{20}	10^{21}	10^{22}	10^{23}	10^{24}	10^{25} и более
P_k	$3,42 \cdot 10^{-9}$	$3,42 \cdot 10^{-7}$	$3,42 \cdot 10^{-5}$	$3,42 \cdot 10^{-3}$	0,29	≈ 1

Таблица 5

K	10-10^{19}	10^{20}	10^{21}	10^{22}	10^{23}	10^{24}
P_k	≈ 0	$7{,}59 \cdot 10^{-19}$	$7{,}59 \cdot 10^{-17}$	$7{,}59 \cdot 10^{-15}$	$7{,}59 \cdot 10^{-13}$	$7{,}59 \cdot 10^{-11}$

K	10^{25}	10^{26}	10^{27}	10^{28}	10^{29}	10^{30} и более
P_k	$7{,}59 \cdot 10^{-9}$	$7{,}59 \cdot 10^{-7}$	$7{,}59 \cdot 10^{-5}$	$7{,}59 \cdot 10^{-3}$	0,549	≈ 1

Таблица 6

K	10-10^{29}	10^{30}	10^{31}	10^{32}	10^{33}	10^{34}
P_k	≈ 0	$4{,}31 \cdot 10^{-18}$	$4{,}31 \cdot 10^{-16}$	$4{,}31 \cdot 10^{-14}$	$4{,}31 \cdot 10^{-12}$	$4{,}31 \cdot 10^{-10}$

K	10^{35}	10^{36}	10^{37}	10^{38}	10^{39}	10^{40} и более
P_k	$4{,}31 \cdot 10^{-8}$	$4{,}31 \cdot 10^{-6}$	$4{,}31 \cdot 10^{-4}$	$4{,}31 \cdot 10^{-2}$	0,99	≈ 1

Таблица 7

K	10-10^{48}	10^{49}	10^{50}	10^{51}	10^{52}	10^{53}
P_k	≈ 0	$1{,}3 \cdot 10^{-18}$	$1{,}3 \cdot 10^{-16}$	$1{,}3 \cdot 10^{-14}$	$1{,}3 \cdot 10^{-12}$	$1{,}3 \cdot 10^{-10}$

K	10^{54}	10^{55}	10^{56}	10^{57}	10^{58}	10^{59} и более
P_k	$1{,}3 \cdot 10^{-8}$	$1{,}3 \cdot 10^{-6}$	$1{,}3 \cdot 10^{-4}$	$1{,}3 \cdot 10^{-2}$	0,72	≈ 1

Таблица 8

K	10-10^{67}	10^{68}	10^{69}	10^{70}	10^{71}	10^{72}
P_k	≈ 0	$3{,}2 \cdot 10^{-19}$	$3{,}7 \cdot 10^{-17}$	$3{,}7 \cdot 10^{-15}$	$3{,}7 \cdot 10^{-13}$	$3{,}7 \cdot 10^{-11}$

K	10^{73}	10^{74}	10^{75}	10^{76}	10^{77}	10^{78} и более
P_k	$3{,}7 \cdot 10^{-9}$	$3{,}7 \cdot 10^{-7}$	$3{,}7 \cdot 10^{-5}$	$3{,}7 \cdot 10^{-3}$	0,31	≈ 1

3 Статистическая безопасность гамм шифрующих

Исследование статистической безопасности гаммы проведено с использованием пакета NIST STS [5].

В 1999 году специалистами NIST в рамках проекта AES (Advanced Encryption Standard) был разработан набор статистических тестов NIST STS (NIST Statistical Test Suite) и предложена методика проведения статистического тестирования ГСЧ (ГПСЧ) [6], которые на настоящий момент наилучшим образом отвечают потребностям всех заинтересованных сторон.

Для принятия решения о прохождении последовательностью случайных (псевдослучайных) чисел статистического теста в пакете NIST STS заложен следующий критерий принятия решения[5].

Пусть дана двоичная последовательность $S = \{s_1, s_2, \ldots, s_n\}$, $s_i \in \{0,1\}$ длиной n бит. Необходимо принять решение, проходит данная последовательность статистический тест на случайность, равновероятность и независимость или нет .

В NIST STS построение критерия принятия решения опирается на вычисление для статистики теста $c(S)$ соответствующего значения вероятности P. Здесь статистика теста рассматривается как реализация случайной величины, которая подчиняется известному закону распределения. Статистка теста строится таким образом, чтобы её большие значения указы-

вали на какой-либо дефект случайности последовательности. Значение вероятности P есть вероятность того, что статистика теста примет значение большее, чем наблюдаемое при опыте в предположении случайности последовательности. Следовательно малые значения P ($P < 0{,}05$ или $P < 0{,}01$) интерпретируются как доказательство того, что последовательность не случайна. Решающее правило формулируется так: *для фиксированного уровня значимости* α *двоичная последовательность* S *не проходит статистический тест, если значение вероятности* $P < \alpha$. Значения α рекомендуется выбирать из интервала [0,001, 0,01].

Пакет NIST STS включает в себя 16 статистических тестов, которые разработаны для проверки гипотезы о случайности двоичных последовательностей произвольной длины, порождаемых генераторами случайных последовательностей или генераторами псевдослучайных последовательностей. Все тесты направлены на выявление различных дефектов случайности. Основным принципом тестирования является проверка нулевой гипотезы H_0, заключающейся в том, что тестируемая последовательность является случайной. Альтернативной гипотезой H_a является гипотеза о том, что тестируемая последовательность не случайна. По результатам применения каждого теста нулевая гипотеза либо принимается, либо отвергается. Решение о том, будет ли заданная последовательность нулей и единиц случайной или нет, принимается по совокупности результатов всех тестов.

Тестирование отдельной двоичной последовательности S с использованием NIST STS выполнялось следующим образом.

Выдвигается нулевая гипотеза H_0– предположение о том, что данная двоичная последовательность S случайна.

По последовательности S вычисляется статистика теста $c(S)$.

С использованием специальной функции и статистики теста вычисляется значение вероятности $P=f(c(S))$, $P \in [0,1]$.

Значение вероятности P сравнивается с уровнем значимости α, $\alpha \in [0{,}001, 0{,}01]$. Если $P \geq \alpha$, то гипотеза H_0 принимается. В противном случае принимается альтернативная гипотеза.

Как уже было сказано, пакет включает в себя 16 статистических тестов. Но фактически в зависимости от входных параметров вычисляется 189 значений вероятности P, которые можно рассматривать как результат работы отдельных тестов.

В результате тестирования двоичной последовательности формируется вектор значений вероятности $\mathbf{P} = \{P_1, P_2, \ldots, P_{189}\}$. Анализ составляющих P_i данного вектора позволяет указать на конкретные дефекты случайности тестируемой последовательности.

Рассмотрим более подробно каждый тест и дефекты последовательности, которые может выявить данный тест.

1. Частотный (монобитный) тест (одно значение вероятности) – выявляет дефект слишком большого количества нулей или единиц в последовательности.
2. Частотный тест для блока (одно значение вероятности) – локализованные отклонения частоты появления единиц в блоке от идеального значения ½.
3. Проверка серий (одно значение вероятности) – слишком быстрая или слишком медленная перемена знака в ходе генерации последовательности.
4. Проверка максимальной длины серии в блоке (одно значение вероятности) – отклонение от теоретического закона распределения максимальных длин серий единиц.
5. Проверка ранга двоичной матрицы (одно значение вероятности) – отклонение эмпирического закона распределения значений рангов матрицы от теоретического, что указывает на зависимость символов в последовательности.
6. Спектральный тест на основе дискретного преобразования Фурье (одно значение вероятности) – отклонение эмпирического закона распределения значений рангов матрицы от теоретического, что указывает на зависимость символов в последовательности.

7. Проверка перекрывающихся шаблонов (одно значение вероятности) – большое количество m- битных серий из единиц в последовательности.
8. Проверка перекрывающихся шаблонов (148 значений вероятности (при длине шаблона 9 бит)) – большое количество заданных непериодических шаблонов в последовательности.
9. Универсальный тест Маурера (одно значение вероятности) – сжимаемость последовательности.
10. Проверка сжатия по алгоритму Лемпеля-Зива (одно значение вероятности) – большая степень сжатия тестируемой последовательности по сравнению с ожидаемой степенью сжатия для случайной последовательности.
11. Последовательный тест (2 значения вероятности) – неравномерность распределения m-битных слов в последовательности.
12. Энтропийный тест (одно значение вероятности) – неравномерность распределения m-битных слов в последовательности (регулярность свойств источника).
13. Проверка накопленных сумм (2 значения вероятности) – большое значение единиц или нулей вначале или в конце двоичной последовательности.
14. Проверка случайных отклонений (8 значений вероятности) – отклонение от теоретического закона распределения попаданий в конкретное состояние при случайном блуждании.
15. Проверка случайных отклонений (18 значений вероятности) – отклонение от теоретического ожидаемого общего количества попаданий при случайном блуждании в заданное состояние.
16. Проверка линейной сложности (одно значение вероятности) – отклонение эмпирического распределения длин эквивалентных ЛРР для последовательности фиксированной длины от теоретического закона распределения для случайной последовательности, что указывает на недостаточную сложность тестируемой последовательности.

4 Результаты экспериментальных исследований

С использованием пакета NIST STS проведено тестирование 9 последовательностей, генерация которых выполнена при помощи функций KDF, внутренняя структура которых основывается на различных алгоритмах (SHA-1, SHA-2, Rijndael, ГОСТ-28.147-89, ГОСТ-34.11-95) .

Для осуществления тестирования были выбраны следующие параметры:

1. Длина тестируемой последовательности $n = 10^6$ бит.

2. Количество тестируемых последовательностей $m = 100$. Таким образом, объем тестируемой выборки составил $N=10^6\times100=10^8$ бит.

3. Уровень значимости $\alpha=0,01$.

4. Количество тестов $q=189$. Таким образом, статистический портрет генератора содержит 18900 значений вероятности P.

В идеальном случае при $m=100$ и $\alpha=0,01$ может быть отвергнута только одна последовательность из ста, т.е. коэффициент прохождения каждого теста должен составлять 99%. Но это слишком жесткое правило. Поэтому и применяется правило на основе доверительного интервала для r_j. Нижняя граница в этом случае составит значение $r_{min} = 0,96015$.

Проведенное тестирование позволяет сделать следующие выводы. Гарантировано полное прохождение всех тестов обеспечивается только KDF на основе функци SHA-2 с длиной блока 512 битов (рис. 3). Последовательности, сформированные с использованием функции хэширования SHA-2 с длиной блока 256 или 384 бита и функции SHA-1 имеют по одному из тестов долю прохождения 0,95 (необходимо 0,96), т.е. значения, близкие к требуемым. При использовании ГОСТ 28147-89 и ГОСТ 34.311-95 по одному из тестов вероятность прохож-

дения составляет 0,94. На рис. 4 приведен статистический портрет последовательностей, сформированных KDF на основе алгоритма Rijndael с длиной блока 256 бит. Как видно из рис. 4, по двум тестам вероятность прохождения составляет 0,95, для длины блока 128 бит ситуация ещё хуже: один тест – меньше 0,94, другой чуть больше. Среди функций KDF на основе Rijndael наиболее приемлемым по статистической безопасности является вариант с 192-битной длиной блока: в этом случае лишь один тест даёт вероятность меньше необходимой (0,95).

Рис. 3

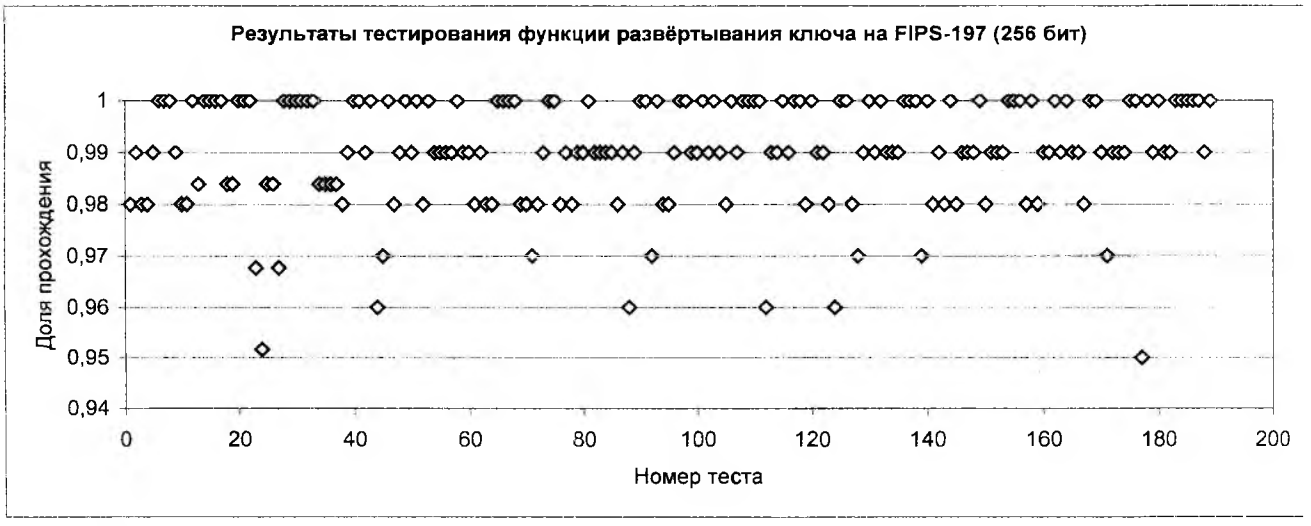

Рис. 4

Как видно из представленных выше рисунков, полное прохождение всех тестов (с коэффициентом прохождения 96%) обеспечивает только функция развёртывания ключа на основе хэш-функции SHA-2 в режиме 512 бит. Функции на основе остальных хэш-функций и симметричных алгоритмов шифрования не проходят по заданному коэффициенту как минимум в одном тесте, хотя все они имеют вероятностные портреты.

Заключение

Схемы шифрования, базирующиеся на примитивах Диффи-Хеллмана, обеспечивают высокую скорость формирования гамм шифрования и поточное шифрование. Наиболее сложным является этап формирования общего секрета, который требует одного возведения в степень по модулю при преобразовании в простом поле или одного скалярного умножения в группе точек эллиптической кривой. Если эти вычисления выполнять предварительно или на сопроцессоре, то скорость поточного шифрования (зашифрование/расшифрование) определяется скоростью хэширования или блочного шифрования.

Вероятности появления коллизий на выходе KDF зависят в определяющей мере от длины блока хэш-функции или блочного шифрования. С увеличением длины блока вероятность коллизии уменьшается соответственно. Приемлемые значения P_k достигаются с использованием функции хэширования SHA-2, в том числе и в перспективе на десятки лет. Приведенные результаты могут быть достигнуты вычислением хеш-функции или блочного шифрования из общего секрета, дополнительных данных и значений счётчиков числа сформированных блоков.

Наилучшую статистическую безопасность обеспечивают KDF, реализованные на основе функции хэширования SHA-2 и использовании дополнительных данных и/или значений счётчиков блоков. Так, последовательности, сформированные с использованием SHA-2, имеют долю прохождения по всем тестам не менее 96%, что позволяет считать их случайными, равновероятными и независимыми.

Список литературы: 1. *О.В. Вербицький.* Вступ до криптології. Львів: ВНТЛ, 1998. 2. X9.42 – 1998, Public Key Cryptography for The Financial Service Industry : Agreement of Symmetric Keys on Using Diffie-Hellman and MQV Algorithms. 3. X9.63 Public Key Cryptography For The Financial Services Industry: Key Agreement and Key Transport Using Elliptic Curve Cryptography, 1999. 207 с. 4. *А.Д. Тевяшев, Ю.И. Горбенко.* Оценка опасности криптоаналитических атак методом создания коллизий // Радиотехника: Всеукр.межвед. науч. техн. сб. 2002. Вып. 126. С. 125 – 131. 5. *Потий А.В., Орлова С.Ю.* Статистическое тестирование генераторов случайных и псевдослучайных чисел с использованием набора статистических тестов NIST STS // Там же. С. 144 – 150. 6. *J. Soto* Randomness Testing of the Advanced Encryption Candidate Algorithms // NIST, 1999.

Харьковский национальный
университет радиоэлектроники

Поступила в редколлегию 16.04.2003