9.John D. Howard An Analysis of Security Incidents on the Internet 1989 -1995.

10. Krsul Ivan Victor. Software Vulnerability Analysis (PhD_thesis) Purdue University 1998.

11. Landwehr Carl E., Bull Alan R., McDermott John P. and William S.Choi. // A Taxonomy of Computer Security Flaws, with Examples. 93-94.

12. http://www.webappsec.org/projects/threat/v1/WASC-TC-v1_0.rus.txt

УДК 681.3.07

А.Я.Белецкий, А.А.Белецкий

СИММЕТРИЧНЫЙ БЛОЧНЫЙ *RSB*-32 КРИПТОАЛГОРИТМ

Современные методы шифрования представляют собой математические преобразования (алгоритмы), в которых сообщения рассматриваются как числа или алгебраические элементы в некотором пространстве [1, 2]. Эти алгоритмы отображают область «осмысленных сообщений» (исходный текст) в область «бессмысленных сообщений» (выходной или шифротекст, шифрограмма). С позиций теории сигналов и процессов *зашифрование* исходного (коррелированного, избыточного, сжимаемого) текста состоит в его «обеливании», т.е. обращении в некоррелированную последовательность символов (элементов) шифрограммы (практически несжимаемой) с плотностью распределения вероятностей элементов выходного алфавита максимально близкой к равномерной.

Для того чтобы иметь возможность восстановить информацию, шифрующие преобразования должны быть обратимыми. Обратное преобразование шифрограммы называется *расшифрованием*. Алгоритмы шифрования параметризуются с помощью криптографических ключей. Совокупность алгоритмов зашифрования и расшифрования, а также описание формата сообщений (входного открытого текста) и пространства ключей образуют криптографическую систему, или *криптосистему*. В том случае, когда для зашифрования и расшифрования используется один и тот же ключ (или ключ расшифрования достаточно легко вычисляется из ключа зашифрования), то такие криптосистемы называются *криптосистемами с секретным* (симметричным) *ключом*.

В данной статье предлагается достаточно гибкая к изменению параметров шифрования (размеров ключей, блоков и элементов замены в блоках) симметричная блочная криптосистема, названная системой ***RSB*-32**. Аббревиатура ***RSB*** происходит от ключевых слов ***Round, Step, Bloc*** – подчеркивая тем самым, что основными для криптоалгоритма являются раундовые преобразования ($\boldsymbol{R}$), разбитые на определенное число шагов ($\boldsymbol{S}$), а действие алгоритма осуществляется над блоками ($\boldsymbol{B}$) открытого или закрытого текстов, причем размер раундового ключа (как элемента общего ключа) составляет 32 бита.

Общее описание *RSB* криптосистемы. ***RSB*** – это итерационный блочный шифр, который доставляет уникальную возможностью по изменению как размеров блоков и ключей, так и размеров элементов блоков, над которыми (элементами) выполняются нелинейные операции замен. В отличие от большинства известных симметричных шифров в ***RSB*** криптосистеме таблицы (матрицы) замен остаются не постоянными, а изменяются в зависимости от состояния секретного ключа.

Основные параметры ***RSB*** шифра:

- Длина раундового ключа - 32 бита.
- Длина общего (шагового) ключа: $r*64$, $r = 1, 2, ...$
- Число шагов шифрования: $s = 1, 2, ...$
- Число раундов шифрования: $r*s$.
- Размер блока: 256, 512, 1024 бита.
- Размер элементов замены: 8, 16, или 32 бита, т.е. ***RSB*** криптосистема

ориентирована на работу с 8–,16–, или 32–разрядными процессорами.

Общий ключ (*Common Key*) в ***RSB*** шифре образуется конкатенацией (объединением) r 32–разрядных раундовых ключей (*Round Key*). Обобщенная структурная схема ***RSB*** алгоритма показана на рис. 1.

Рис.1

Данная схема отображает процесс преобразования текстов как для алгоритма зашифрования, так и расшифрования (естественно – с учетом выполнения требований обратимости преобразований). Поэтому в дальнейшем мы ограничимся в основном пояснением организации процесса зашифрования открытого текста.

Перед началом процедуры зашифрования входной открытый текст разбивается на блоки, размер которых может быть выбран равным 256, 512 или 1024 бита. Если последний блок оказался меньше выбранного размера, то он дополняется (пробелами) до полного блока. Назовем такой текст *расширенным файлом*. Объем расширенного файла в ходе зашифрования не меняется, поэтому объем шифротекста всегда будет кратным размеру блока.

Развернутая структурная схема ***RSB*** алгоритма в режиме зашифрования приведена на рис.23, в котором использованы такие обозначения:

- RC (*Round Code*) – операции зашифрования текста раундовым ключом (подключом общего ключа);
- RK_{ij} – j -й раундовый ключ на i -м шаге зашифрования.

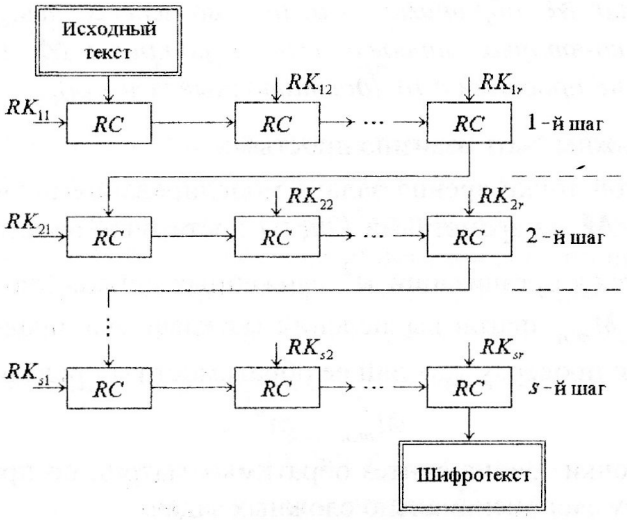

Рис.2

Таким образом, ***RSB*** алгоритм (как и большинство современных симметричных блочных шифров) состоит из большого количества повторяющихся преобразований – раундов. Как следует из структурной схемы криптоалгоритма (рис. 2), сначала производятся последовательные преобразования всех блоков расширенного файла раундовым ключом RK_1, затем ключом RK_2 и, наконец, ключом RK_r. На этом заканчивается обработка текста на первом шаге зашифрования. При условии, что число шагов шифрования s больше единицы, происходит частичное обновление раундовых ключей за счет циклического (кругового) сдвига на семь разрядов общего ключа CK (рис. 3) и описанная выше процедура преобразования повторяется на очередном шаге шифрования.

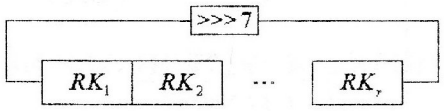

Рис.3

Математическую основу преобразований в ***RSB*** криптоалгоритмах составляют так называемые модульные матричные преобразования

$$\boldsymbol{y} = (\boldsymbol{x}\,\boldsymbol{M})_m \qquad (1)$$

где

$$b = (a)_m$$

– основная операция модульной арифметики, которая означает вычисление остатка b от деления целого числа a на модуль (основание системы счисления) m.

В соотношении (1) через $\boldsymbol{x}$ и $\boldsymbol{y}$ обозначены входная и выходная кодовые комбинации (вектор–строки) соответственно, а $\boldsymbol{M}$ – так называемая *модульная матрица* преобразования. К модульным будем относить квадратные матрицы n-го порядка, все элементы которых a_{ij} являются неотрицательными целыми числами, причем

$$a_{ij} \in Z_m = \{0, 1, ..., m-1\}, \quad i, j = \overline{1, n}\ .$$

Если существует модульная матрица $\overline{\boldsymbol{M}}$, обратная $\boldsymbol{M}$, то из равенства (1) получим

$$\boldsymbol{x} = (\boldsymbol{y}\,\overline{\boldsymbol{M}})_m\ .$$

Следовательно, модульные матричные преобразования обратимы, если обратима матрица, участвующая в этом преобразовании. Сформулируем необходимые и достаточные условия обратимости модульных матриц.

Модульная матрица $\boldsymbol{M}$ обратима, если она, во-первых, невырождена (необходимые условия) и, во-вторых, определитель Δ матрицы $\boldsymbol{M}$ по модулю m является числом взаимно простым с m (достаточные условия),

т.е. числа $(\Delta)_m$ и m должны быть взаимно простыми.

С алгоритмической точки зрения задача моделирования полного множества обратимых модульных матриц $\boldsymbol{M}$ с параметрами (m, n) достаточно простая и в программной реализации на ЭВМ сводится к организации n^2 вложенных циклов (по числу элементов матрицы $\boldsymbol{M}$). Общее число $\mathcal{M}_{m,n}$ шагов вычислений (каждый шаг предполагает формирование *матрицы претендента* и проверку условий ее обратимости) определяется соотношением:

$$\mathcal{M}_{m,n} = m^{n^2}\ ,$$

т.е. с вычислительной точки зрения синтез обратимых матриц по приведенному выше алгоритму относится к классу экспоненциально сложных задач.

Естественно, что, во-первых, моделирование и хранение полного множества модульных обратимых матриц с параметрами $(m, n) \geq 4$ потребовало бы чрезвычайно больших вычислительных ресурсов и, во-вторых, нет никакой необходимости в таком изобилии матриц. Поэтому в качестве модульных обратимых матриц в разрабатываемой криптосистеме выбраны так называемые *индикаторные матрицы* систем Виленкина–Крестенсона функций (ВКФ) [3]. Их особенность состоит в том, что они являются *правосторонне* симметрическими матрицами, т.е. симметричными относительно вспомогательной диагонали. Синтез таких матриц, например, четвертого порядка $(n = 4)$ может быть организован следующей последовательностью вложенных циклов для моделирования десяти свободных элементов матриц (оставшиеся шесть элементов матриц являются связанными условием правосторонней симметрии)

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 3 \\ 8 & 9 & 6 & 2 \\ 10 & 8 & 5 & 1 \end{bmatrix}. \qquad (2)$$

В схеме ранжирования (2) цифра 1 означает, что перебор элемента a_{11} матрицы $\boldsymbol{M}$ управляется внешним циклом, тогда как цифра 10, отвечающая элементу a_{41}, соответствует внутреннему циклу программы моделирования. Для предлагаемой схемы формирования обратимых модульных матриц

$$M_{4,4} = 458752.$$

Предложенная схема моделирования обратимых модульных матриц не является единственной. Таким же свойством обратимости обладают *левосторонне* симметрические индикаторные матрицы ВКФ, а также ряд других матриц, не являющихся симметрическими.

Преобразование содержимого блоков шифруемого расширенного текста осуществляется под управлением раундового ключа, секторное разбиение которого приведено на рис. 4.

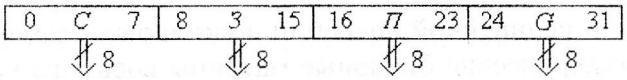

Рис. 4

Согласно данному рисунку, 32-разрядный раундовый ключ разбивается на четыре сектора (байта), которые управляют следующими операциями преобразования:

C – сдвиг циклический блока;

$З$ – нелинейная замена элементов блока;

$П$ – перестановка битов в пределах блока;

G – гаммирование.

Как следует из рис. 4, адресные шины, которыми извлекаются матрицы замен, перестановок и гаммирования, являются восьмиразрядными. Это означает, что для организации указанных операций достаточно хранить в базе 256 прямых $\boldsymbol{M}$ и обратных $\overline{\boldsymbol{M}}$ матриц преобразования соответственно. В то же время мощность массива индикаторных матриц систем ВКФ, моделируемых по схеме (2), составляет 458752, т.е. является избыточной. Обратим внимание на то, что множество $M_{4,4}$ содержит так называемые «слабые» матрицы. К *слабым* будем относить матрицы, операнды преобразований которых x и y *абсолютно* или *частично линейно связаны*.

К абсолютно линейно связанным (АЛС) будем относить единичные матрицы $\boldsymbol{E}$ и матрицы инверсной перестановки $\boldsymbol{I}$, а также матрицы $\boldsymbol{E}$ и $\boldsymbol{I}$, умноженные на число $k = 3$, являющееся взаимно простым с основанием $m = 4$. К частично линейно связанным (ЧЛС)

будем относить индикаторные матрицы из массива $M_{4,4}$, для которых найдется хотя бы одна пара строк a и b, поразрядная сумма содержимых которых (a) и (b), представленных в форме десятичных чисел, удовлетворяет условию

$$(a)+(b)=(a+b)\ . \qquad (3)$$

Рассмотрим пример ЧЛС матрицы

$$\boldsymbol{M}=\begin{matrix}1\\2\\3\\4\end{matrix}\begin{bmatrix}0&0&1&0\\0&1&0&1\\1&0&1&0\\1&1&0&0\end{bmatrix}. \qquad (4)$$

Как следует из (4), условие (3) выполняется для следующих пар строк матриц: (1,2), (1,4), (2,3) и (3,4), т.е. матрица (4) является ЧЛС матрицей.

И, наконец, к нелинейно связанным (НЛС) будем относить обратимые модульные матрицы преобразования, для которых равенство (3) не соблюдается. Примером такой матрицы может служить матрица

$$\begin{bmatrix}0&1&0&1\\0&1&1&0\\1&0&1&1\\1&1&0&0\end{bmatrix}.$$

В ***RSB*-32** шифре стохастический отбор 256 матриц прямого преобразования, используемых для реализации операций линейных перестановок битов в блоках и гаммирования, осуществляется из массива $M_{4,4}$, из которого предварительно удаляются все АЛС и ЧЛС матрицы. Для организации нелинейной операции подстановки (замены) элементов блока используются дважды симметрические бинарные матрицы восьмого порядка, правило формирования которых приведено ниже в соответствующем разделе статьи.

Перейдем к описанию основных операций шифрования.

Операция циклического сдвига (C). Посредством данной операции осуществляется стохастический круговой сдвиг влево (для операции зашифрования) N-разрядного блока ($N=256$, 512 или 1024 бита) на случайное число разрядов, которое лежит в пределах от 1 до 255. Семь старших разрядов этого числа считываются из сектора C раундового ключа (см. рис. 4), а в младший разряд байта принудительно заносится единица. Тем самым байт, которым определяется порядок циклического сдвига блока, будет содержать нечетное число в пределах от 1 до 255.

Операция замены ($З$) является обратимой нелинейной матричной операцией и служит для рассеивания элементов алфавита шифрограммы. Поясним сначала алгоритм операции нелинейной замены (подстановки) для восьмиразрядных элементов блока. Подстановка

$$\boldsymbol{x}=\{x_7, x_6, ..., x_0\}\rightarrow \boldsymbol{y}=\{y_7, y_6, ..., y_0\}$$

представляет собой нелинейную замену байтов, которая выполняется независимо для каждого входного байта шифруемого блока. Матрицы подстановки, с помощью которых составляются $\boldsymbol{S}$-блоки, являются инвертируемыми матрицами и образуются в результате композиции двух преобразований:

1. Получением мультипликативного обратного элемента $\boldsymbol{x}^{-1}$ над расширенным конечным полем Галуа $GF(2^8)$, которое строится над кольцом многочленов по модулю неприводимого многочлена

$$\varphi(x) = x^8 + x^4 + x^3 + x + 1,$$

при этом нулевой элемент переходит сам в себя.

2. Выполнением афинного преобразования над примитивным двоичным полем Галуа $GF(2)$, которое задается следующим образом:

$$\boldsymbol{y} = \boldsymbol{x}^{-1}\boldsymbol{M} + \boldsymbol{\beta}, \qquad (5)$$

где $\boldsymbol{M}$ – невырожденная двоичная матрица восьмого порядка; $\boldsymbol{\beta}$ – восьмиразрядная вектор-строка, выбираемая из сектора 3 раундового ключа RG (см. рис. 4).

Схема формирования матриц $\boldsymbol{M}$ задана соотношением

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & 9 & 10 & 11 & 12 & 13 & 14 & 7 \\ 3 & 10 & 15 & 16 & 17 & 18 & 13 & 6 \\ 4 & 11 & 16 & 19 & 20 & 17 & 12 & 5 \\ 5 & 12 & 17 & 20 & 19 & 16 & 11 & 4 \\ 6 & 13 & 18 & 17 & 16 & 15 & 10 & 3 \\ 7 & 14 & 13 & 12 & 11 & 10 & 9 & 2 \\ 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \end{bmatrix}, \qquad (6)$$

т.е. $\boldsymbol{M}$ составляется как дважды симметрическая матрица восьмого порядка. Как показали результаты моделирования на ЭВМ, всего существует 2144 невырожденных матриц (6) с тремя или пятью нулями в каждой строке.

Следует отметить, что нелинейность преобразования (5) обеспечивается исключительно нелинейностью инверсии $\boldsymbol{x}^{-1}$. Если бы это преобразование применялось непосредственно к элементу $\boldsymbol{x}$, афинное уравнение (5) было бы абсолютно линейным.

Поясним алгоритм нелинейной замены 16-разрядных элементов $\boldsymbol{x}$. Сначала по обычным правилам определяются мультипликативные обратные элементы $\boldsymbol{x}$ над полем $GF(2^{16})$, которые затем представляются восьмиразрядными четверичными векторами. Далее выполняется операция (5), после чего вектор $\boldsymbol{y}$ переводятся в двоичную форму. Аналогичным образом осуществляется нелинейная замена 32 - разрядных элементов $\boldsymbol{x}$, причем в преобразовании (5) операнд $\boldsymbol{x}$ представлется восьмиразрядным 16-ричным числом.

Операция перестановки (Π) битов в блоке реализуется соотношением (1), в котором $m = 4$, а $\boldsymbol{M}$ является матрицей перестановки ($\boldsymbol{M}_{\Pi}$), извлекаемой из массива обратимых модульных матриц четвертого порядка по случайному адресу, который задается восьмиразрядной шиной сектора Π раундового ключа.

Рассмотрим схему перестановок битов для 256 – разрядного блока. Организуем восьмиразрядный двоичный счетчик и зафиксируем его состояние b_i ($i = \overline{0, 7}$) на некотором j-м шаге счета. Преобразуем исходный вектор $\boldsymbol{b}_j$, которым задается восьмиразрядный двоичный номер замещаемого бита, в четырехразрядный четверичный вектор $\boldsymbol{x}$ по схеме, заданной соотношением:

b_7 b_6	b_5 b_4	b_3 b_2	b_1 b_0
x_3	x_2	x_1	x_0

. (7)

После этого выполняется преобразование (1), завершающее вычисление номера разряда $\boldsymbol{y}$, в который перемещается бит, находившийся в разряде блока под номером $\boldsymbol{x}$.

Для организации перестановок битов в 512 – разрядном блоке он (блок) предварительно разбивается на два подблока по 256 бит, а затем включается выше описанная схема перестановки битов 256 – разрядного блока. Возможны различные варианты разбиения блока на полублоки. В качестве первого из них можно предложить простое деление исходного блока пополам. Во втором варианте в первый полублок можно собрать все биты с четными (нечетными) номерами разрядов исходного блока, а оставшиеся биты – во второй полублок и т.д. Аналогичным образом можно произвести перестановку битов и в 1024 – разрядном блоке.

Операция гаммирования (G) подобна операции добавления раундового ключа (*Add Round Key*) в шифре ***AES*** [1] и осуществляется посредством простого поразрядного *XOR* блока и *расширенного раундового ключа* (*Key Expansion*), который мы будем называть *гамма функцией* и обозначать $\boldsymbol{G}$. Функция $\boldsymbol{G}$ представляет собой бинарный вектор, размер которого совпадает с размером блока N, и составляется по схеме

$$\boldsymbol{G} = \boldsymbol{G}_0 \,||\, \boldsymbol{G}_1 \,||\, \boldsymbol{G}_2 \,||\, ... \,||\, \boldsymbol{G}_{k-1}\,, \quad k = N/32,$$

где $||$ есть знак конкатенаци.

Вектор-строка $\boldsymbol{G}_0$ образуется в результате поразрядного сложения по модулю 2 текущего раундового ключа и 32-разрядного вектора, формируемого из строк четверичной матрицы $\boldsymbol{M}$, извлекаемых из базы данных по адресу, находящемуся в секторе $\boldsymbol{G}$ раундового ключа. Для $i \geq 1$

$$\boldsymbol{G}_i = \boldsymbol{G}_{i-1} \overset{2}{\oplus} \boldsymbol{G}^*_{i-1}\,,$$

где $\boldsymbol{G}^*_j$ – кодовая комбинация, образуемая из вектор-строки $\boldsymbol{G}_j$ в результате ее циклического сдвига на 7 разрядов влево.

Функции шифрования. В ***RSB*** системе предусмотрены следующие функции шифрования, реализуемые раундовыми ключами (табл. 1).

Таблица 1

№ п/п	Название функции	Обозначение
1	Базовая	$B/b/z$
2	Расширенная	$R/b/z$
3	Стохастическая	$S/b/z$

В табл. 1 приняты такие обозначения: большие буквы определяют один из трех $\{B, R, S\}$ вариантов шифрования; малые буквы b и z задают размеры блока и элемента замены в блоке соответственно. Ниже даны примеры полного обозначения ***RSB*** криптосистем: ***RSB***-$32S/512/32$; ***RSB***-$32R/1024/8$ и т.д.

Перейдем к описанию раундовых функций шифрования (F), воспользовавшись дополнительными символами, которыми обозначаются такие преобразования:

- $\widetilde{C}$ – стохастический круговой сдвиг расширенного файла;
- $\widetilde{З}$ – операция нелинейной замены, при которой каждый элемент блока замещается своей индивидуальной матрицей замены $\boldsymbol{M}_З$.

Операции $С$, $П$ и $З$ обсуждались в предыдущих разделах и нам остается лишь уточнить схемы выполнения операций $\widetilde{C}$ и $\widetilde{З}$. Стохастический сдвиг расширенного файла (операция $\widetilde{C}$) осуществляется в конце каждого шага преобразования под управлением двоичного байта, семь старших разрядов которого отбираются из сектора $П_2$ раундового ключа RK_r, а в младший разряд записывается единица. Тем самым оператором $\widetilde{C}$ расширенный

файл сдвигается влево (при зашифровании) или вправо (при расшифровании) на нечетное число разрядов от 1 до 255. Операция $\widetilde{C}$ введена для того, чтобы разрушить жесткое разбиение текста на блоки, что усложняет работу криптоаналитика при дешифровании закрытого текста.

Алгоритмы формирования восьмиразрядных адресов (A), с помощью которых из массива обратимых модульных матриц извлекаются матрицы замены $\boldsymbol{M}_3$, индивидуальные для каждого элемента блока (функция $\widetilde{З}$), представлены в табл. 2.

Таблица 2

Размер элемента замены (n)	Размер блока (N)		
	256	512	1024
8	$RK\frac{32}{1}$	$2 \circ RK\frac{32}{1}$	$4 \circ RK\frac{32}{1}$
16	$RK\frac{16}{2}$	$RK\frac{32}{1}$	$2 \circ RK\frac{32}{1}$
32	$RK\frac{8}{4}$	$RK\frac{16}{2}$	$RK\frac{32}{1}$

В данной таблице принята такая схема записи алгоритмов: $RK\frac{l}{k}$ означает, что раундовый ключ RK подвергается l-кратному циклическому сдвигу влево (при зашифровании) или вправо (при расшифровании) на k разрядов в каждом такте сдвига. После выполнения l шагов кругового сдвига раундовый ключ восстанавливает свое исходное состояние, поскольку $l \cdot k = 32$, что совпадает с размером раундового ключа RK. Каждым тактом сдвига раундового ключа RK осуществляется модификация состояния адресной шины ($З$) и, тем самым, на очередном i-м такте i-й элемент блока замещается своей индивидуальной матрицей $\boldsymbol{M}_{3i}$.

Для размера блока $N = 512$ и величине элемента замены $n = 8$ каждый блок шифруемого текста содержит по $N / n = 64$ элементов замены. В то же время 32-разрядный раундовый ключ дает возможность сформировать не более 32-х различных адресов (при круговом сдвиге на один разряд), по которым извлекаются матрицы замен. Поэтому 512-разрядный блок разбивается на две равные части и на каждом такте сдвига происходит замена двух элементов блока (по одному в обеих половинах разбиения). При $N = 1024$ и $n = 8$ производится деление блока на четыре части, т.е. каждой извлекаемой из массива матрицей $\boldsymbol{M}_3$ осуществляется замена четырех элементов блока. Приведенная схема замен элементов блоков обозначена в табл. 2 множителем $k \circ$ ($k = 2, 4$), который поставлен перед аббревиатурой RK.

Приведем символическое описание раундовых функций (F) для вариантов зашифрования B, R и S соответственно:

$$FB = СЗПG;$$
$$FR = С\widetilde{З}ПG;$$
$$FS = (С\widetilde{З}ПG)\widetilde{C}.$$

Каждый вариант шифрования в ***RSB*** криптосистеме может быть реализован в так называемой *матричной форме*, доставляющей повышение скорости обработки текстов, но с потерей гибкости смены параметров шифрования. В этом случае идентификатор системы дополняется буквой M. Например, ***RSB***-$32R/256/16/M$.

Идея матричной (скоростной) формы шифрования состоит в следующем. Выберем, для примера, вариант шифрования ***RSB***-$32R/256/16/M$ с параметрами $r = 6$ и $s = 8$, т.е. процесс шифрования производится над 256-разрядными блоками расширенного текста (открытого или зашифрованного) секретным 192-разрядным ключом (составленным из шести

32-разрядных раундовых ключей). Размер элементов (слов) замены равен 16 бит, а весь процесс шифрования завершается за 8 шагов (итераций).

Рассмотрим выполняемые процедуры для алгоритма зашифрования. Раундовая функция зашифрования имеет вид: $FR = C\widetilde{З}ПG$, которой предполагается, что каждый блок расширенного шифруемого текста подвергается сначала стохастическому круговому сдвигу (C), после этого последовательно все 16-разрядные элементы блока замещаются своими индивидуальными матрицами замены, затем выполняется операция перестановки битов в блоке ($П$) и, наконец, проводится гаммирование блока (G). Когда заканчивается обработка последнего блока расширенного файла, аналогичная обработка блоков проводится сначала вторым, затем третьим и, наконец, шестым раундовым ключом. После этого раундовые ключи обновляются (за счет циклического сдвига влево на семь разрядов общего ключа) и процедура обработки блоков продолжается по выше описанной схеме на протяжении восьми шагов шифрования.

Поскольку секретный ключ шифрования известен субъектам обмена закрытой информации, то это дает возможность заранее извлечь из базы данных все необходимые модульные матрицы, участвующие в операциях перестановок и замен, а также зафиксировать порядки стохастических круговых сдвигов блоков для всех 48-ми (так как $r = 6$ и $s = 8$) раундов зашифрования. Тем самым предоставляется возможность освободить систему от многих ресурсозатратных операций и повысить как скорость зашифрования, так и расшифрования.

И в заключении раздела отметим, что ***RSB*** криптосистема допускает возможность работы в различных широко используемых режимах, таких как CBC (*Ciphentext Block Chaining* – сцепление блоков шифротекста), OFB (*Output Feedback* – обратной связи по выходу) и др.

Статистический анализ эффективности *RSB* криптоалгоритма проводился с помощью программого пакета ***NIST STS*** (версия 1.8), специально разработанного Институтом Стандартов и Технологий США для оценки качества криптографических методов и датчиков псевдослучайных чисел [6]. Широкое внедрение ***NIST STS*** подтверждает исключительную эффективность данной системы анализа. Набор тестов ***NIST STS*** применялся практически для всех промышленных стандартов криптоалгоритмов. ***NIST STS*** использует 15 основных статистических тестов, которые, на самом деле представляют из себя 188 испытаний, для определения соответствующей глобальной или локальной оценки.

Ниже приведен статистический портрет (рис. 6) програмной реализации блочного криптоалгоритма *RSB* (длинна ключа: 256, число шагов шифрования: 32, размер блока: 256).

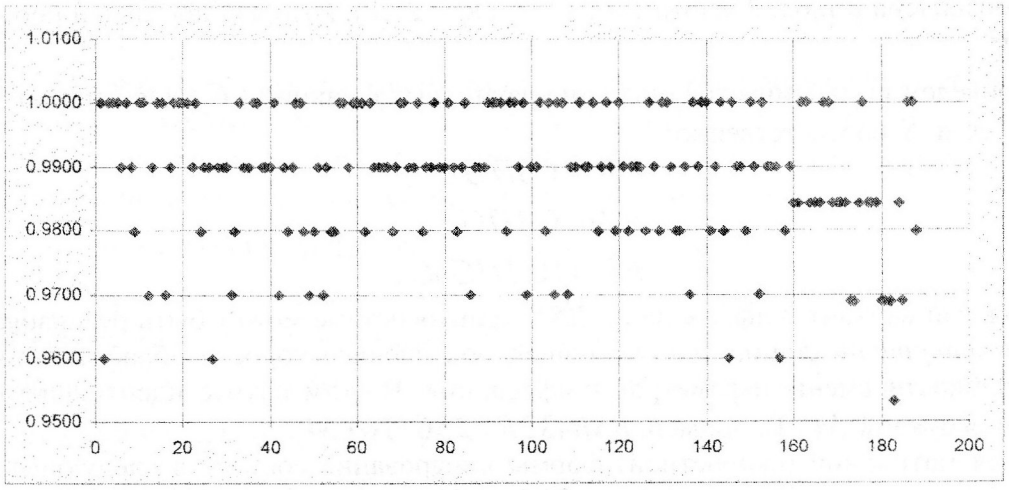

Рис. 6

Для сравнения приведем (табл. 3) статистические оценки других известных критпографических алгоримов.

Таблица 3

Генератор	Количество тестов, в которых тестирование прошло больше 99% последовательностей	Количество тестов, в которых тестирование прошло больше 96% последовательностей	Количество, тестов в которых значение вероятности		
			p<=0.01	p<= 0.001	p<=0.05
RSB	130	187	0	0	6
ГОСТ 28147-98	130	186	1	0	3
DES	125	188	1	0	16
IDEA	125	187	1	0	7
AES	125	188	2	0	8

В качестве открытого текста для всех генераторов был выбран один и тот же 17-Мбайтный русскоязычный текст (словарь Даля).

Выводы

1. Шифр RSB имеет четкую и ясную структуру, за один раунд в нем преобразуются все биты входного блока, в отличие от шифра Фейстеля (являющегося основой большинства криптоалгоритмов), где за один раунд изменяется, как правило, лишь половина бит входного блока.

2. Как показали результаты статистических испытаний ***RSB*** криптосистемы для 256 – разрядного ключа ($r = 8$) и 32–х итераций шифрования, эффективность алгоритма зашифрования, оцениваемая количеством тестов в пакете ***NIST STS***, в котором тестирование прошло больше 99% и соответственно 96 % последовательностей, оказалось на уровне российского алгоритма ***ГОСТ*** 28147-89 и превосходит эффективность широко используемых стандартов криптографической защиты, таких как ***DES***, ***IDEA*** и ***AES*** (*Rijndael*).

Список литературы

1.Мао В. Современная криптография. Теория и практика. – М.: «Вильямс», 2005. – 768с.

2.Шнайер Б. Прикладная криптография. –М.: «ТРИУМФ», 2003. – 816 с.

3.Белецкий А.Я. Комбинаторика кодов Грея. – Киев: КВІЦ, 2003. – 506 с.

4.Харин Ю.С. Берник В.И., Матвеев Г.В., Агиевич С.В. Математические и компьютерные основы криптологии. – Мн.: Новое знание, 2003. – 382 с.

5.Молдавян Н.А., Молдавян А.А., Еремееев М.А. Криптография: от примитивов к синтезу алгоритмов. – СПб.: БХВ, 2004. – 448 с.

6.Random Number Generation and Testing. http: www.csrc.nist.gov/rng/

УДК 681.04

О.С. Петров, Є.О.Валуйський

ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ ІНФОРМАЦІЇ В ПРОЦЕСІ ДИСТАНЦІЙНОГО НАВЧАННЯ НА ОСНОВІ ТЕХНОЛОГІЇ СМАРТ-КАРТ.

Вступ

Для забезпечення більш ефективного захисту інформації, аутентифікації користувачів, цілісного контролювання процесу дистанційного навчання, потрібно розробити адекватну модель верифікації. Необхідно забезпечити аутентичнісить користувача, на рівні єдиного документу (студентський, читацький квиток) для доступу до електронних ресурсів (бібліотека, комплект учбово-методичної документації тощо), послуг Інтернет, а також забезпечення цілісності інформації в процесі передачі й прийому даних через відкриті мережі при тестуванні студента в процесі дистанційного навчання.