Тогда

$$X = \begin{vmatrix} 4 & 8 & 5 & 4 \\ 11 & 9 & 6 & 12 \\ 5 & 10 & 1 & 2 \\ 9 & 5 & 7 & 1 \end{vmatrix}.$$

Для проверки проводим вычисления, записывая только те строки и столбцы, значения в которых полностью определяются заданными квадратами:

$$[X, y; 1] = \begin{vmatrix} 6 & 12 & 12 & 5 \\ 6 & 12 & 5 & 10 \\ 9 & 5 & 7 & 1 \end{vmatrix}, \quad [X, y; 2] = \begin{vmatrix} 0 & 0 & 7 & 8 \\ 10 & 7 & 11 & 9 \end{vmatrix}, \quad [X, y; 3] = \begin{vmatrix} 3 & 6 & 9 & 12 \end{vmatrix},$$

$$[X, y; 1, z; 3]_{1,1} = 12, \quad [X, y; 2, z; 2]_{1,1} = 7, \quad [X, y; 3, z; 1]_{1,1} = 3.$$

ЛИТЕРАТУРА

1. *Романьков В. А.* Введение в криптографию. М.: Форум, 2012.

УДК 519.17 DOI 10.17223/2226308X/13/18

ЭКСПЕРИМЕНТАЛЬНАЯ ОЦЕНКА ПРОИЗВОДИТЕЛЬНОСТИ ОДНОГО КЛАССА КРИПТОАЛГОРИТМОВ НА ОСНОВЕ ОБОБЩЕНИЯ СЕТЕЙ ФЕЙСТЕЛЯ

В. М. Фомичёв, Д. А. Бобровский, А. М. Коренева

Представлены результаты экспериментальных исследований производительности алгоритма 256-3 (с блоком 256 бит и тремя функциями обратной связи), предложенного российскими исследователями в 2018 г. Производительность 256-3 оценивается величиной 24,57 циклов на байт. Проведено сравнение с известными блочными шифрами, получены оценки для программных реализаций алгоритмов на языке программирования C++ с использованием библиотеки Crypto++. Установлено, что производительность 256-3 от 1,2 до 2,6 раз превышает производительность алгоритмов «Магма» (ГОСТ 28147-89), «Кузнечик» (ГОСТ 34.12-2018), SEED, HIGHT, Camellia-256, Kalyna-256/256, MARS-256, CAST-256, что указывает на положительные (с позиции синтеза) эксплуатационные качества алгоритма 256-3.

Ключевые слова: *блочные шифры, производительность шифрования, 256-3, ГОСТ 28147-89, ГОСТ 34.12-2018, «Магма», «Кузнечик», AES, Rijndael, SEED, SM4, HIGHT, Camellia, Kalyna, MARS, CAST, RC6, Crypto++.*

Введение

Развитие информационных технологий и необходимость защиты информации определяют актуальность разработки новых криптографических алгоритмов, соответствующих современным требованиям к криптографической стойкости и эксплуатационным качествам. Для обеспечения конфиденциальности информации при её передаче, обработке и хранении требуются алгоритмы с высокой производительностью и варьируемыми параметрами (размерами длины ключа и блока) в зависимости от типа задачи.

С целью увеличения производительности блочного шифрования в [1, 2] исследован класс регистровых преобразований $R(n, r, m)$, реализуемых автономными регистрами сдвига длины n над множеством $V_r = \{0, 1\}^r$ с m обратными связями, $n > m \geqslant 1$. Идея

увеличения производительности состоит в увеличении размера блока данных при относительно небольшом увеличении числа обратных связей. Предложены способы построения биективных раундовых функций с блоками от 256 до 1056 бит, при которых координатные функции шифрующих подстановок нелинейные и реализуют полное перемешивание битов входного блока. На примере алгоритма 256-3 (с блоком 256 бит и тремя функциями обратной связи, аналогичными функции усложнения ГОСТ 28147-89) показано, что построенные алгоритмы превышают по производительности алгоритмы на основе классической сети Фейстеля. Экспериментально установлено, что производительность 32-раундового алгоритма 256-3 в два раза превышает производительность ГОСТ 28147-89.

В работе представлены новые результаты экспериментальных исследований производительности алгоритма 256-3, проведено сравнение с известными блочными шифрами, которые являются международными, отраслевыми и национальными стандартами, а также рекомендованными международной организацией по стандартизации (ISO) [3].

1. Схема раундовой функции алгоритма 256-3

Опишем принцип построения раундовой функции g алгоритма 256-3 [1, 2] (схема на рис. 1). Для фиксированного раунда обозначим:

$X = (X_0, \ldots, X_7)$ — входной блок раунда, $X \in V_{256}, X_k \in V_{32}, 0 \leqslant k \leqslant 7$;

$Y = (Y_0, \ldots, Y_7)$ — выходной блок раунда, $Y \in V_{256}, Y_k \in V_{32}, 0 \leqslant k \leqslant 7$;

S — сумма по модулю 2^{32} нескольких подблоков входного блока, представленных числами из кольца вычетов $\mathbb{Z}_{2^{32}}$;

q_j — раундовый ключ, использующийся при вычислении значения функции обратной связи с номером $j \in \{1, 2, 3\}$;

$\oplus$ — сложение по модулю 2, $\boxplus$ — сложение по модулю 2^{32}.

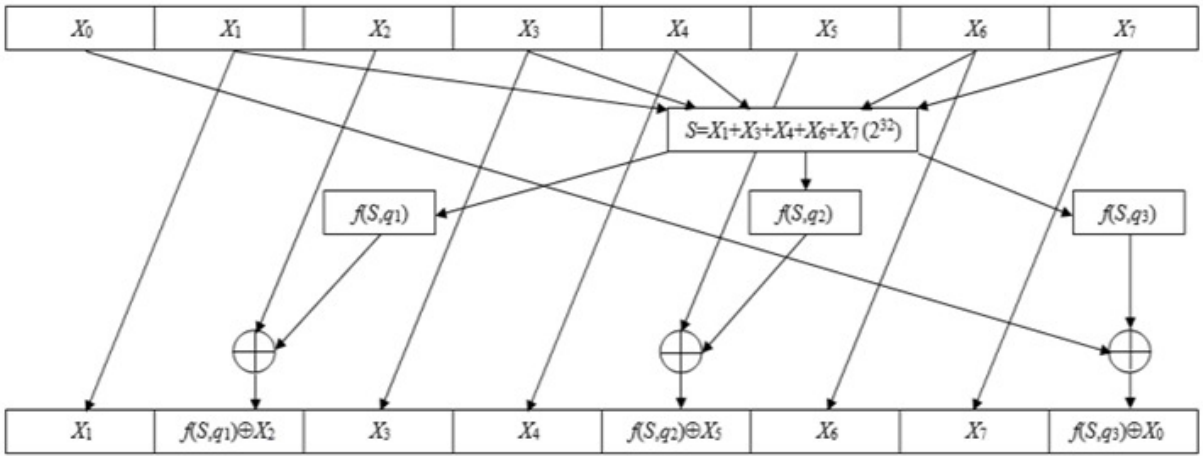

Рис. 1. Схема раундовой подстановки g алгоритма 256-3

Раундовая подстановка g алгоритма 256-3 и обратная к ней определены формулами

$$g(q_1, q_2, q_3)(X_0, \ldots, X_7) = (X_1, f(S, q_1) \oplus X_2, X_3, X_4, f(S, q_2) \oplus X_5, X_6, X_7, f(S, q_3) \oplus X_0),$$
$$g^{-1}(q_1, q_2, q_3)(Y_0, \ldots, Y_7) = (f(S', q_3) \oplus Y_7, Y_0, f(S', q_1) \oplus Y_1, Y_2, Y_3, f(S', q_2) \oplus Y_4, Y_5, Y_6),$$

где $S = X_1 \boxplus X_3 \boxplus X_4 \boxplus X_6 \boxplus X_7$ и $S' = Y_0 \boxplus Y_2 \boxplus Y_3 \boxplus Y_5 \boxplus Y_6$.

Регистр сдвига g имеет три идентичных обратных связи $X_2 \oplus f(S, q_1)$, $X_5 \oplus f(S, q_2)$, $X_0 \oplus f(S, q_3)$, каждая из которых построена по принципу раундовой функции блочного

шифра «Магма» (ГОСТ 34.12-2018). Функция f имеет вид $f(S, q_j) = T^{11}(W_{8,4}(S \boxplus q_j))$, где $\boxplus q_j$ — сложение с раундовым ключом q_j по модулю 2^{32}; $W_{8,4}$ — преобразование V_{32}, реализуемое восемью 4-битовыми s-боксами алгоритма «Магма»; T^{11} — преобразование циклического левого сдвига на 11 бит.

2. Описание эксперимента и результаты исследования

Проведено сравнение производительности алгоритма 256-3 с производительностью известных блочных шифров: AES-256, «Магма», «Кузнечик», SEED, SM4, HIGHT, Camellia-256, Kalyna-256/256, MARS-256, CAST-256, RC6-256. Программная реализация алгоритмов шифрования 256-3 и «Кузнечик» выполнена на языке программирования C++ с использованием кроссплатформенной бесплатной библиотеки Crypto++ 8.2 с открытым исходным кодом [4]. Для остальных алгоритмов использованы реализации из библиотеки Crypto++. Выбор данной библиотеки обусловлен большим количеством криптографических алгоритмов и высокой скоростью реализаций в сравнении с другими криптографическими библиотеками (на языках Python, C, C ++).

Эксперименты проведены на ПЭВМ с процессором Intel(R) Core(TM) i5-7600 с постоянной тактовой частотой $U = 3{,}89\,\text{ГГц}$, архитектура операционной системы 64-битная (x64). Расширение системы команд AES-NI отключено. Оптимизация программного кода — /O2.

Для каждого из алгоритмов выполнено зашифрование открытого текста длиной $L = 268435456$ байт в режиме простой замены, ключ вырабатывался программным датчиком случайных чисел в составе библиотеки Crypto++. Время t, затраченное на зашифрование, измерялось с помощью системных часов реального времени стандартной библиотеки chrono. Затем рассчитывалось количество мебибайт (2^{30} байт) обработанного открытого текста в секунду (МиБ/с) и независимая от частоты процессора характеристика производительности шифра — количество циклов на байт (CpB), согласно формуле $\text{CpB} = tU/L$. Результаты экспериментов приведены в таблице в порядке убывания производительности.

Производительность шифров

Шифр	Число раундов	CpB	МиБ/с
AES-256	14	7,92497	468,114
SM4	32	14,6612	253,035
RC6-256	20	15,2273	243,628
256-3	32	24,5674	151,005
MARS	32	29,2092	127,008
Camellia-256	24	31,247	118,725
CAST-256	48	31,4735	117,871
«Магма»	32	48,2291	76,9202
Kalyna-256/256	14	50,7198	73,1429
SEED	16	55,022	67,4239
«Кузнечик»	10	62,2676	59,5782
HIGHT	32	64,3055	57,6901

Выводы

Результаты показали, что производительность реализации 256-3 ниже производительности AES-256, SM4, RC6-256 в 3, 1,68 и 1,61 раз соответственно. Это связано с высокой скоростью выполнения примитивных операций, заложенных в данные алгоритмы, на системах с архитектурой Intel IA-64. В то же время производительность

256-3 в 1,2–2,6 раз превышает производительность алгоритмов «Магма» (ГОСТ 34.12-2018), «Кузнечик» (ГОСТ 34.12-2018), SEED, HIGHT, Camellia-256, Kalyna-256/256, MARS-256, CAST-256, что указывает на положительные (с позиции синтеза) эксплуатационные качества алгоритма 256-3 и представляет данный алгоритм перспективным для потенциального применения в программных и аппаратных средствах защиты информации.

ЛИТЕРАТУРА

1. *Fomichev V. and Koreneva A.* Encryption performance and security of certain wide block ciphers // J. Comput. Virol. Hack. Tech. 2020. https://doi.org/10.1007/s11416-020-00351-1
2. *Fomichev V. M., Koreneva A. M., Miftahutdinova A. R., and Zadorozhniy D. I.* Evaluation of the maximum performance of block encryption algorithms // Math. Aspects Cryptogr. 2019. V. 10. No. 2. P. 7–16.
3. ISO/IEC 18033-3. IT Security Techniques. Encryption Algorithms. P. 3: Block Ciphers. https://www.iso.org/standard/54531.html.
4. Криптографическая кроссплатформенная C++ библиотека Crypto++ 8.2 с открытым исходным кодом. https://www.cryptopp.com/

УДК 519.17 DOI 10.17223/2226308X/13/19

ХАРАКТЕРИСТИКИ АЛГОРИТМА КОНТРОЛЯ ЦЕЛОСТНОСТИ ДАННЫХ НА ОСНОВЕ АДДИТИВНЫХ ГЕНЕРАТОРОВ И s-БОКСОВ

В. М. Фомичев, А. М. Коренева, Т. Р. Набиев

При проведении анализа программного обеспечения актуальна задача контроля целостности данных больших массивов, при решении которой важно обеспечить приемлемый компромисс между криптографическими свойствами алгоритма контроля целостности и ресурсами, необходимыми для его реализации. Для блоков данных размера 1 кбайт (1024 байта) предложен алгоритм генерации 128-битового кода контроля целостности (ККЦ) с положительными (с позиции синтеза) эксплуатационными и криптографическими свойствами. Алгоритм построен на основе преобразований аддитивных генераторов и s-боксов и реализует функцию $\psi(g^t)\colon V_{2^{13}} \to V_{128}$ со свойством полного перемешивания входных данных. При $6 \leqslant t \leqslant 100$ каждый бит кода существенно зависит от всех битов информационного блока. При случайном равновероятном выборе начального состояния u вероятность получить любой код Q оценивается величиной 2^{-128}. Среднее число опробований пар блоков (u, u'), где $u \neq u'$ и $Q(u) = Q(u')$, приблизительно равно 2^{64}. Сложность вычисления функции $\psi(g^t)$ имеет порядок $t(5u + 8v)$, где u — вычислительная сложность суммирования двух чисел по модулю 2^{64}; v — сложность вычисления s-бокса. В соответствии с проведёнными экспериментами скорость генерации ККЦ варьируется в пределах от 3500 ($t = 6$) до 250 Мбит/с ($t = 96$), соответственно при тех же значениях t время генерации ККЦ варьируется в пределах от 18 до 250 мкс.

Ключевые слова: *аддитивные генераторы, контроль целостности, матрично-графовый подход, перемешивающие свойства, регистры сдвига.*

Введение

Одной из важных задач защиты информации является контроль целостности, который осуществляется с помощью присоединения создателем информации к информа-