Байт-ориентированный алгоритм поточного шифрования на основе совокупности равномерно плотных блоков нелинейной подстановки

Белецкий А. Я.[1]

[1]Проф., д.т.н., кафедра электроники, Национальный авиационный университет,
пр. Космонавта Комарова,1, Киев, Украина, abelnau@ukr.net

***Аннотация* — Рассматриваются вопросы построения нового байт-ориентированного алгоритма поточного шифрования, в котором шифрующая гамма-последовательность стохастических битов формируется совокупностью равномерно плотных примитивов нелинейной подстановки. Равномерно плотными названы такие примитивы, отклики которых равномерно распределены на диаграмме рассеяния. Проведен сравнительный анализ эффективности разработанного криптопреобразования и Rijndael-алгоритма в режиме поточного шифрования. Обсуждаются направления применения предлагаемых шифров в различных приложениях.**

***Ключевые слова:* алгоритмы синхронного поточного шифрования, равномерно плотные блоки нелинейной подстановки, криптографическая защита информации.**

Byte-oriented Algorithm Stream Ciphering Based on the Aggregate Uniformly Dense Blocks Nonlinear Substitution

Beletsky A. Ja.[1]

[1]Prof., Dr. Sc., Department of Electronics, National Aviation University,
pr. Kosmonavt Komarov, 1, Kiev, Ukraine, abelnau@ukr.net

***Abstract* — Questions of construction of the new byte-oriented stream encryption algorithm, which encrypts the gamma-stochastic sequence of bits formed by a set of primitives uniformly dense nonlinear substitution. Evenly dense are called primitives, responses which are evenly distributed on the scatterplot. A comparative analysis of the efficiency of the developed and cryptographic transformation algorithm Rijndael-stream encryption mode. The directions of application of the proposed codes in various applications.**

***Keywords:* synchronous stream encryption algorithms, uniformly dense blocks nonlinear substitution, cryptographic protection of information.**

Введение

Различают два основных класса алгоритмов шифрования: блочные и поточные. В *блочных шифрах* в результате криптопреобразования двух одинаковых блоков открытого текста образуются два одинаковых блока шифрованного текста. Избежать этого позволяют *поточные шифры* [1], в которых шифрующее преобразование «элемента» открытого текста меняется от одного элемента к другому. Такой эффект прослеживается, например, в блочных DES и AES шифрах, которые в режиме сцепления блоков фактически преобразуются в поточные шифры.

На практике термин поточный шифр используют, как правило, только в том случае, когда «элементы» открытого текста очень малы и составляют один бит или один байт. Если шифруемым элементом является бит, то такие поточные шифры называют *бит-ориентированными шифрами*. Если же шифруемым элементом служит байт, то шифры называют *байт-ориентированными*. Реже встречаются поточные шифры, размер шифруемых элементов в которых превышает байт.

Большинство поточных шифров могут быть названы *двоичными аддитивными шифрами*. В таких шифрах $k-$битный секретный ключ K используется только для управления генератором, порождающего *псевдослучайную последовательность* (ПСП) битов $k_0, k_1, \ldots, k_{N-1}$, называемую *ключевым потоком* K, где $N >> k$. Шифртекст C образуется путем сложения по модулю 2 битов T_i открытого текста T и битов k_i ключевого потока K, в результате чего приходим к алгоритму шифрования

$$C_i = T_i \oplus k_i, \qquad i = 0,1,\ldots,N-1.$$

Дешифрование криптотекста C выполняется подобно алгоритму шифрования, т.е. $T_i = C_i \oplus k_i$.

Поточные шифры находят применение в тех случаях, когда требуется высокая скорость передачи информации, например, при трансляции «живого» видео, в системах сотовой связи и др., или при передаче по каналам связи массивов данных большого объема.

Основные требования, предъявляемые к шифрующим ПСП, таковы: большой *период гаммы*; «хорошие» *статистические свойства,* оцениваемые, в основном, пакетами тестов *NIST STS* [2] и *DIEHARD* [3]; соблюдение *постулатов Голомба* [4]; высокая *линейная сложность* последовательности [5] и др.

Одним из важнейших в перечисленных выше показателях поточных шифров является *период шифрующей гаммы.* Подлинно случайные ПСП должны быть бесконечными. Но величина периода никак не отображает характер распределения битов в последовательности. Поэтому, на данный момент, для определения свойств гаммирующей последовательности применяют пакеты статистического тестирования. При этом, как утверждает Д. Кнут, чем большим числом пакетов подтверждены успешные прохождения тестов, тем выше уверенность конструктора поточного шифра в качестве разработки.

Целью данного доклада является изложение результатов разработки новых байт-ориентированных поточных шифров, в которых гамма-последовательность битов формируется группой *равномерно плотных* блоков нелинейной подстановки (РП-примитивов), осуществляющих преобразования «байт-в-байт».

Равномерно плотными будем называть такие примитивы ($S-$блоки), отклики которых равномерно распределены на диаграмме рассеяния.

Стохастический синтез РП-примитивов

Классическим примитивом нелинейной подстановки (ПНП) может быть назван $S-$блок симметричного блочного AES шифра (алгоритм Rijndael), осуществляющий преобразование

$$y = x_f^{-1} \cdot A + b. \qquad (1)$$

Диаграмма рассеяния $S-$блока AES шифра представлена на рис. 1. Как следует из визуального осмотра рис. 1, диаграмма рассеяния $S-$блока шифра AES далека от диаграммы с равномерной плотностью. При том, что среднее число точек $\bar{n}$ в элементах разбиения диаграммы рассеяния (порядок которых составляет 8x8) должно равняться четырем, фактически в элементы попадают от нуля до десяти точек. СКО σ числа точек, содержащихся в элементах диаграммы рассеяния, равно 5.12, а коэффициент корреляции составляет $r = -0.0438$.

Ниже кратко изложен один из вариантов стохастического синтеза равномерно плотных ПНП «байт-в-байт», который назовем *алгоритмом синтеза S-блоков с выбыванием*. Суть алгоритма состоит в следующем.

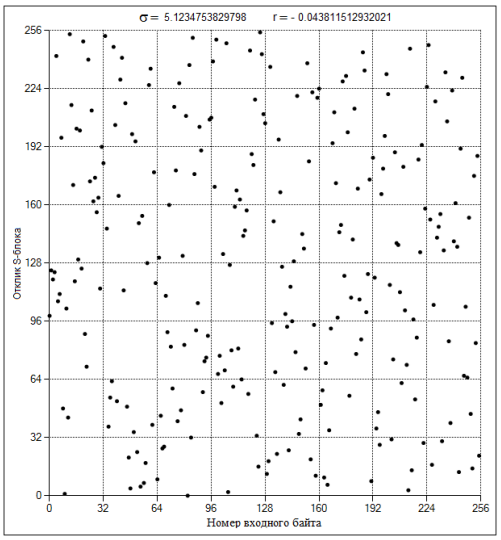

Рисунок. 1. Диаграмма рассеяния примитиванелинейной подстановки алгоритма Rijndael

Пусть X и $Y-$ входной и выходной байты стохастического $S-$блока соответственно, десятичные эквиваленты которых лежат в диапазоне от 0 до 255. Разобьем шкалы диаграмм рассеяния X и Y, поименованные на рис. 1 как «Номер входного байта» и «Отклик $S-$блока», на восемь равномерных интервалов, каждый из которых содержит по 32 целочисленных отсчета. Стохастическое заполнение квадратов будем осуществлять таким способом, чтобы каждый квадрат разбиения $S_{k,l}$, $k,l = \overline{1,8}$, где $k-$ номер строки, а $l-$ номер столбца диаграммы рассеяния, содержал по четыре точки с координатами $(X,Y) \in S_{k,l}$, причем квадрат $S_{1,1}$ разместим в окрестности начала координат диаграммы.

Условимся считать, что если точка (X,Y) располагается на левой боковой грани квадрата $S_{k,l}$ или на его основании, то она принадлежит этому квадрату. Выберем, как наиболее простой, порядок заполнения диаграммы рассеяния по столбцам. Образуем два вспомогательных вектора длины 256 байт, а именно, вектор $\boldsymbol{X}$, в ячейки которого внесены восьмибитные числа от 0 до 255 и вектор $\boldsymbol{Y}$, все ячейки которого обнулены.

Сначала генерируют случайное равномерно распределенное в интервале от 0 до 1 число (РРЧ) x и вычисляют ординату $y_0 = [x \cdot 256]$, где $[a]-$ целая часть значения a.

Байт числа y_0 размещается в нулевой ячейке вектора $\boldsymbol{Y}$, т.е. в ячейке $\boldsymbol{Y}(0)$, а ячейка $\boldsymbol{X}(y_0)$ исключается из вектора $\boldsymbol{X}$. Затем генерируется очередное РРЧ x и вычисляется ордината $y_1 = [x \cdot 255]$. Число, содержащееся в ячейке $\boldsymbol{X}(y_1)$,

размещается в ячейке $\boldsymbol{Y}(1)$, а ячейка $\boldsymbol{X}(y_1)$ исключается из вектора $\boldsymbol{X}$ и т.д. Следовательно, на $i-$м шаге генерации РРЧ x в $i-$ю ячейку вектора $\boldsymbol{Y}$ записывается байт числа $y_i = [x \cdot N_i]$, где $N_i = 256 - i$, а также исключается ячейка $\boldsymbol{X}(y_1)$ вектора $\boldsymbol{X}$.

Если на некотором $i-$м шаге генерации РРЧ x окажется, что в каком-либо квадрате первого столбца диаграммы рассеяния находятся четыре точки, то все оставшиеся 28 ячеек вектора $\boldsymbol{X}$ этого квадрата временно исключаются из рассмотрения.

Перед заполнением квадратов второго столбца таблицы рассеяния восстанавливаются 224 ячейки вектора $\boldsymbol{X}$, т.е. исходный вектор $\boldsymbol{X}$ за исключением тех 32 ячеек, которые были задействованы на этапе формирования первого столбца таблицы. Точно также переход к заполнению каждого очередного столбца диаграммы рассеяния предполагает исключение из вектора $\boldsymbol{X}$ 32 ячеек, использованных при формировании предыдущего столбца таблицы. Стартовое значение индекса i для квадратов второго столбца равно 32, для третьего – 64 и т.д.

Таким образом, после заполнения всех столбцов диаграммы рассеяния вектор $\boldsymbol{Y}$ будет содержать всю информацию относительно стохастически моделируемого примитива нелинейной подстановки. При этом номер ячейки $x = \overline{0,255}$ вектора $\boldsymbol{Y}$ является аргументом, а содержимое ячейки $\boldsymbol{Y}(x) = y \in [0, 255]-$ функцией f нелинейного преобразования $y = f(x)$.

Пример РП $S-$блока показан на рис. 2.

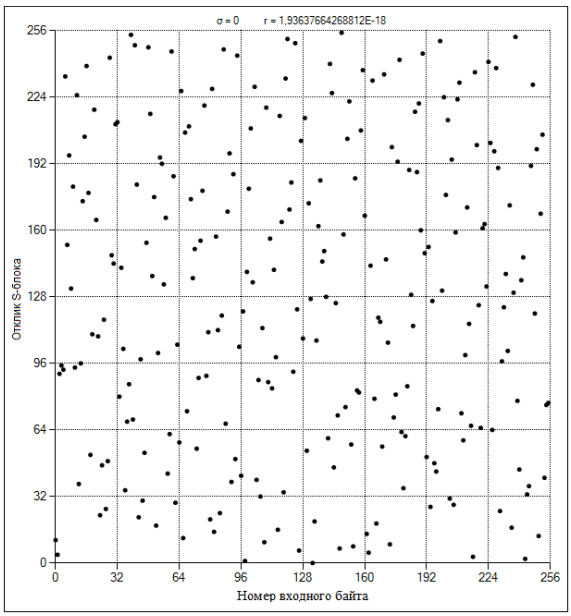

Рисунок. 2. Диаграмма рассеяния РП ПНП

БЛОК ПРЕОБРАЗОВАНИЯ БАЙТОВ (БПБ),

составленный из совокупности n равномерно плотных ПНП, является основным узлом предлагаемого (п. 4) поточного шифра. На рис. 3 показана структурная схема БПБ, в состав которого входят четыре ПНП, обозначенные как $S-$box$i$, $i = \overline{0,3}$, типа «байт-в-байт».

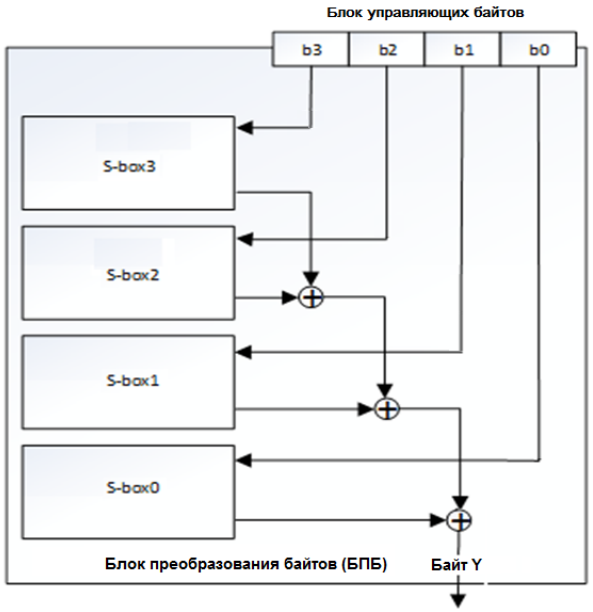

Рисунок 3. Структурная схема четырехбайтного блока преобразования

Обобщенное преобразование, реализуемое БПБ, отображается соотношением:

$$y = S_{n-1}(b_{n-1}) \oplus S_{n-2}(b_{n-2}) \oplus \ldots \oplus S_0(b_0),$$

где $S_i(b_i)-$ отклик $i-$го $S-$блока на входной байт b_i; $\oplus-$ оператор поразрядного сложения по модулю 2 (операция *XOR*).

Выходные байты $\boldsymbol{Y}$ БПБ (рис. 3) могут быть использованы для поточного шифрования последовательности байтов открытого текста T:

$$C_i = T_i \oplus Y_i, \quad i = 0,1,\ldots \qquad (2)$$

Соотношением (2) предполагается, что для каждого i соблюдается неравенство $Y_{i+1} \neq Y_i$.

Операция *XOR* является одной из наипростейших и (при правильном использовании) наиэффективнейших операций шифрования. Для взлома последовательности C_i третьей стороной необходимо знать:

1) Размерность m вектора байтов $\boldsymbol{B} = \{b_i\}$, над которым выполняется преобразование;

2) Непосредственно байты b_i, $i = \overline{0, m-1}$;

3) Стохастические таблицы $S-$блоков.

Допускается, что п. 1) и 2), т.е. размерность вектора n и сам вектор байтов $\boldsymbol{B}$, могут быть открытыми. Если таблицы $S-$блоков являются закрытыми, то шифрование (2) становится не взламываемым. В самом деле, по состоянию на 2016 год лобовая атака ключа длиною 128 бит (16 байт) по методу его последовательного перебора невозможна. Тем более бессмысленной является

восстановление «вслепую» 256-байтного секретного ключа, не говоря уже о том, что как в блоках преобразования (рис.3), так и в поточных шифраторах на их основе (п. 4), таких $S-$блоков может быть несколько, что значительно понижает вероятность взлома шифра.

РП-АЛГОРИТМ ПОТОЧНОГО ШИФРОВАНИЯ

Строится на основе БПБ, рассмотренных в п. 3. Вариант такого шифра показан на рис. 4, в котором обозначено: $\boldsymbol{X}$ – Входной байт; $\boldsymbol{Y}$ – Байт гаммирования; и $\boldsymbol{Z}$ – Выходной байт.

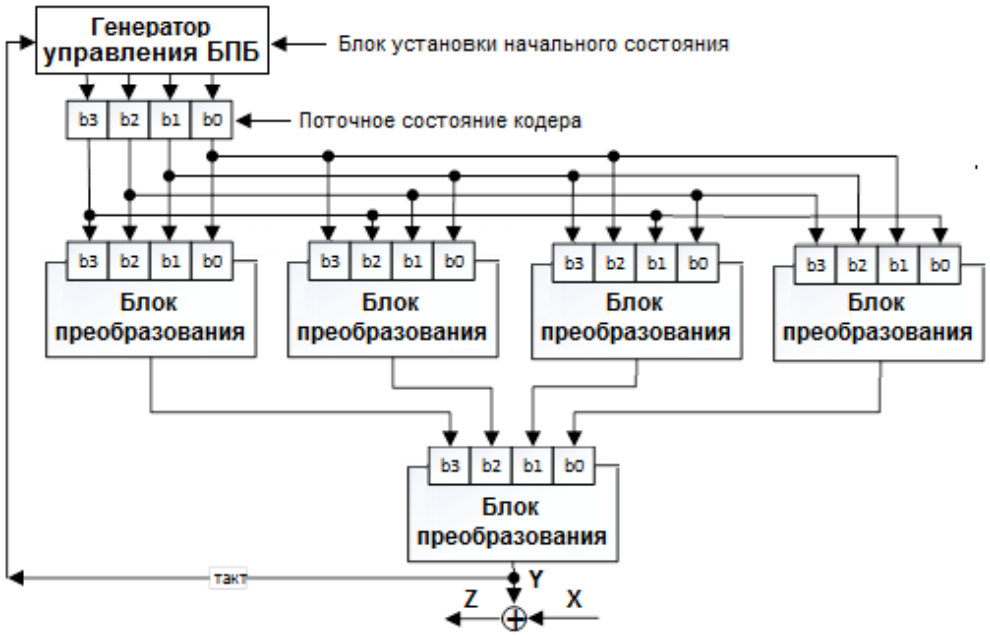

Рисунок 4. Структурно-логическая схема алгоритма поточного шифрования на основе БПБ

В основу протоколов шифрования положены правила, регламентирующие использование криптографических преобразований и алгоритмов в информационных процессах. Обобщенная схема построения криптографического *протокола зашифрования данных* приведена на рис. 5.

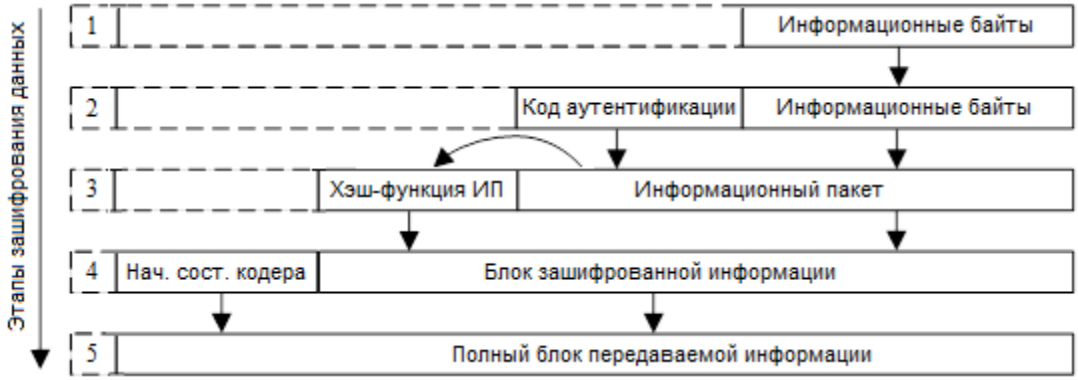

Рисунок 5. Структурно-логическая схема протокола зашифрования данных

Протокол зашифрования решает следующие задачи:

- на этапе (1) формируется *блок информационных байтов*, который
- на этапе (2) дополняется секретным *кодом аутентификации* (в авиационной терминологии – поточным БПБ-шифром (рис. 4). К полученному *блоку зашифрованной информации* (БЗИ) присоединяется блок *начального состояния кодера* (НСК);
- на этапе (5) формируется полный *блок передачи информации* (БПИ) потребителю и, тем индивидуальным кодом «свой-чужой»), совместно образуя *информационный пакет* (ИП);
- на этапе (3) вычисляется хэш-функция (ХФ) информационного пакета, а затем
- на этапе (4) объединенный блок ХФ + ИП подвергается криптографическому преобразованию самым, завершается протокол зашифрования данных.

На приемной стороне этапы расшифрования данных выполняются в последовательности (рис. 6), обратной последовательности этапов зашифрования.

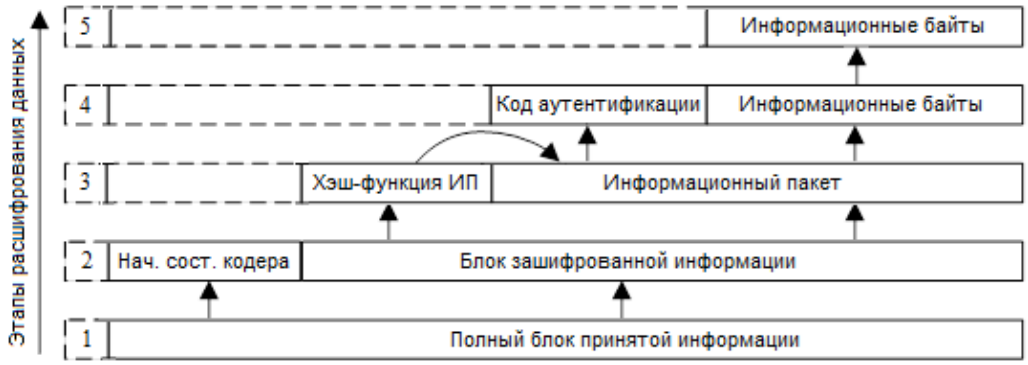

Рисунок 6. Структурно логическая схема протокола расшифрования данных

Протоколом расшифрования данных выполняются следующие преобразования:

- на этапе (1) в процессоре приемника размещается полный блок принятой информации, из которого
- на этапе (2) выделяются блоки НСК и БЗИ;
- на этапе (3) посредством блока НСК запускается процесс расшифровки данных по схеме, показанной на рис. 4, в результате которого из БЗИ выделяются ХФ и ИП. Затем вычисляется ХФ расшифрованного ИП. Если рассчитанная ХФ отличается от принятой, то ИП игнорируется. В случае совпадения ХФ
- на этапе (4) из ИП извлекаются код аутентификации и файл информационных байтов, а последний
- на этапе (5) размещается в регистре блока информационных байтов.

На этом протокол расшифрования данных завершается.

Роль *синхронизирующего элемента* в протоколах обмена данными выполняет содержимое блока установки начального состояния генератора управления БПБ (рис. 4), которое передается по каналу связи в открытом виде, что не нарушает секретности передачи данных, так как третья сторона не в состоянии воспроизвести расшифровующую гамму, поскольку для него (противника) остаются закрытыми $S-$ блоки шифраторов.

Код (или ключ) аутентификации в составе хэш-функции информационного пакета обеспечивает возможность одному оператору НПУ поддерживать связь с целой группой БПЛА. С этой целью каждому аппарату выделяется индивидуальный код аутентификации. В то же время $S-$ блоки шифраторов НПУ, как и всей группы беспилотников, могут быть одинаковыми, что снижает затраты на эксплуатацию системы.

ВЫВОДЫ

Отличительная особенность предлагаемого способа криптографических преобразований командно-телеметрической информации (КТИ) состоит в простоте алгоритмическо-программного обеспечения системы защиты КТИ. Есть все основания предполагать, что разработан новый, не имеющий отечественных и зарубежных аналогов, оригинальный алгоритм поточного шифрования данных, который по критериям быстродействия передачи информации и степени «отбеливания» исходного текста, оцениваемой энтропией шифрограммы, не уступает, а в отдельных машинных экспериментах превышает, соответствующие показатели AES-шифратора.

ЛИТЕРАТУРА REFERENCES

[9] Асосков А. В. Поточные шифры / А. В. Асосков, М. А. Иванов, А. А. Мирский, А. А. Рузин и др. – М.: КУДИЦ-ОБРАЗ, 2003. – 336 с.

[10] A Statistical Test Suite for Random and Psevdorandom Number Generators for Cryptographic Applications. NIST Special Publication 800-22? Rev. 1a / Technology Administration, U.S. Depart. of Commerce. – Washington: NIST. – 2010. – P. 131.

[11] Marsaglia G. DIEHARD Statistical Tests. [Электронный ресурс] – Режим доступа: http://stat.fsu.edu/~geo/diehard.html

[12] Golomb S. W. Shift Register Sequences. / S. W. Golomb. – San Francisco: Holden Day, 1967 (and also reprint: Aegan Park Press, 1982). – ISBN 978-3-540-44523-4

[13] Безверхая Г. С. Анализ перспективы развития поточного шифрования. / Г. С. Безверхая. // Системи обробки інформації, 2009, випуск. 7(81). – С. 54-55.исунок 3. Дерево восьмиточечного БПФ