УДК 004.383 DOI: 10.34759/trd-2020-115-13

Устройство формирования уникальной последовательности, используемой при обезличивании персональных данных

Спеваков А.Г.*, Калуцкий И.В.**

Юго-Западный государственный университет,

ЮЗГУ, ул. 50 лет Октября, 94, Курск, 305040, Россия

**e-mail: aspev@yandex.ru*

***e-mail: kaluckiy_igor@yandex.ru*

Статья поступила 23.11.2020

Аннотация

В статье обосновывается использование устройств формирования уникальной последовательности, для решения задачи повышения быстродействия процесса обезличивания персональных данных и повышения уровня защищенности информационной системы. Предложена математическая модель и алгоритм процесса скоростного формирования уникальной последовательности для каждого субъекта персональных данных, обрабатываемых в информационной системе, предложенным устройством. Приведены структурно-функциональные схемы работы устройства.

Ключевые слова: обезличивание данных, хеширование, преобразование, быстродействующее устройство.

В современных автоматизированных системах обрабатывается огромное количество данных, в том числе, подпадающих под категорию персональных. Такая информация является наиболее уязвимой и подвержена наибольшему количеству атак со стороны злоумышленников. По экспертным оценкам количество киберпреступлений с использованием персональных данных ежегодно увеличивается в разы. Со стороны государственных органов предпринимаются огромные усилия, направленные на предотвращение хищения персональных данных, вводятся дополнительные требования к организации обработки и защите данных [1]. Исходя из Конвенции Совета Европы о защите физических лиц при автоматизированной обработке персональных данных и Федерального закона № 152 «О защите персональных данных», оператор обязан обеспечить конфиденциальность обрабатываемых данных, что ведет к значительным материальным затратам. Для решения таких задач используются методы обезличивания [2]. Так же методы обезличивания могут применяться в различных отраслях народного хозяйства, для повышения защищенности процесса передачи данных. Например, при организации защищенной радиосвязи с летательными аппаратами различного назначения.

Решению задач обезличивания данных посвящены работы [3-7], но их анализ показал, что предложенные подходы преобразования конфиденциальных персональных данных в обезличенную не конфиденциальную последовательность обычно занимает много времени, обладает низкой стойкостью к атакам и имеет ограничения при обработке больших массивов персональных данных с частыми

изменениями, в записях частично сохраняются связи между атрибутами обезличенных данных и соответствующими им атрибутами персональных данных, при изменении значений отдельных атрибутов может произойти только изменение состава данных, а не обезличивание. Это позволило сформулировать основные задачи исследования.

Постановка задачи

Предположим, что исходные данные даны в виде предварительных значений $D_N(d_1, d_2, ..., d_M)$, где M - общее количество атрибутов, а N - количество строк таблицы. Атрибуты d могут быть ключевыми и неключевыми. В результате число значений ключа равно $K\,(0 < K < M)$. При формировании данных для хеширования используется закрытый ключ PK с пригодностью 512. Для заданной последовательности данных функция деперсонализации может быть представлена как задача разбиения данных на два множества A_1 и A_2, причем A_1 содержит конфиденциальные данные, A_2 - анонимную информацию, и нахождения уникальной последовательности d_0, такой, что для набора $F(a_1, a_2, ..., a_n, PK)$ значение будет уникальным. При этом выполняется следующее условие – невозможность обратного преобразования, т.е. найти блок данных из любого d_0 - невозможно, что в свою очередь позволяет установить взаимосвязь элементов первого и второго множеств.

Поэтому технической задачей является разработка метода, алгоритма и быстродействующего устройства формирования уникальной последовательности, для обезличивания персональных данных, используемых для повышения скорости и

конфиденциальности обезличивания данных и позволяющих устранить выявленные недостатки.

Математическая модель устройства формирования уникальной последовательности, используемой при обезличивании персональных данных

Для формирования хеш последовательности используется следующий метод, основанный на принципе криптографической губки [8-10], предусматривающей два основных этапа:

1) Исходное сообщение P подвергается многораундовым перестановкам F, производится накопление и обработка всех блоков сообщения, из которого будет выработан хеш [11].

2) Вывод получившегося в результате перестановок значения Z, выработка хеш-значения и вывод результатов до тех пор, пока не будет достигнута требуемая длина хеша [12] .

В поглощающей фазе сначала задается начальное состояние из нулевого вектора с размером до 1600 бит. Далее проводится операция XOR фрагмента исходного состояния с фрагментом исходного сообщения p_0 с фрагментом исходного состояния размером r, состояния с размером, остальная часть состояния с емкостью остается той же самой.

Результат обрабатывается функцией F, представляющей собой много-раундовую псевдослучайную перестановку и повторяется до исчерпания блоков исходного

сообщения [13,14]. Далее следует фаза сжатия, на которой можно извлечь хэш произвольной длины. Блок-схема алгоритма хеширования представлена на рисунке 1.

Функция F в данном алгоритме выполняет 24 раунда, раунд включает в себя работу пяти функций Theta, Chi. Pi, Rho, Iota, последовательно обрабатывающие внутреннее состояние на каждом раунде [15-17]. Функция Theta представлена следующими выражениями (1):

$$C[x]=A[x,0]\oplus A[x,1]\oplus A[x,2]\oplus A[x,3]\oplus A[x,4],x=0\ldots4$$

$$D[x]=C[x\text{-}1]\oplus(C[x+1]>>>1),x=0\ldots4;$$

$$A[x,y]=A[x,y]\oplus D[x],x=0\ldots4,y=0\ldots4 \quad (1)$$

Функция Chi представлена следующим выражением (2):

$$A[x,y]=B[x,y]\oplus(\sim B[x+1,y]\&B[x+2,y],x=0\ldots4,y=0\ldots4 \quad (2)$$

Функция Pi, Rho представлена следующим выражением (3):

$$B[y,2x+3y]=A[x,y]>>>r(x,y),x=0\ldots4,y=0\ldots4 \quad (3)$$

Функция Lota представлена следующим выражением (4):

$$A[0,0]=A[0,0]\ xor\ RC \quad (4),$$

где B- временный массив, имеющий ту же структуру, что и массив состояний; и являются ли временные массивы, каждый из которых содержит 5 64-битных слов; C и D - массивы, определяющий количество битов для каждого слова состояния.

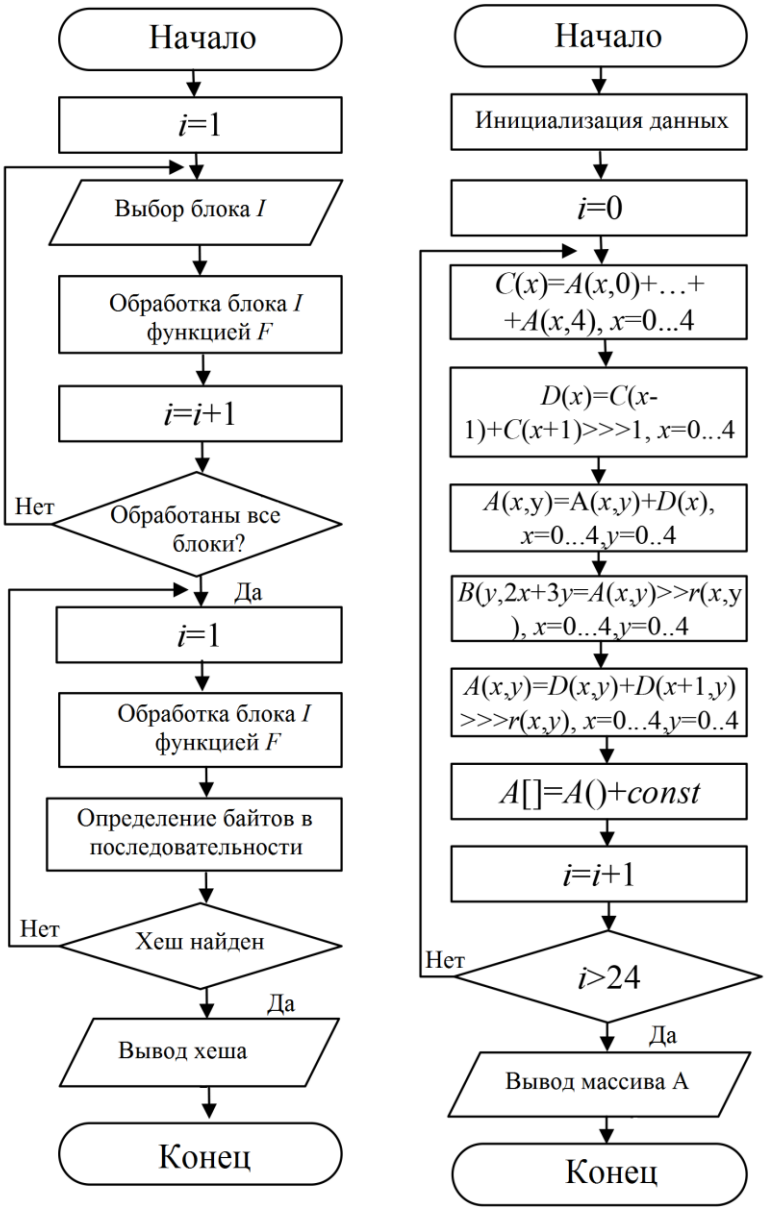

Рисунок 1 - Блок схема алгоритма хеширования

Алгоритм состоит из следующих шагов [18]:

Шаг 1: в начале алгоритма проводится инициализация данных. Размер состояния составляет 1600 бит. Переменной *i* присваивается значение 0.

Шаг 2: после этого начинается обработка массива с функциями $C[x], D[x], A[x, y], B[y,2x+3y], A[x,y]$, и кроме этих операций проводится операция сложение по модулю два константы *RC* со словом $A[0,0]$.

Шаг 3: после обработки данных с помощью подфункций идет проверка количества раундов. Если условие $i > 24$ истинно, то выполняется вывод массива A. Если нет, то инкрементируем *i* на единицу и переходим на Шаг 2.

Новизна разработанной математической модели заключается в вычислении уникальной последовательности данных по которой можно однозначно идентифицировать кортеж из исходной информации, при этом получение исходной информации из хеш идентификатора практически не реализуемая задача [18-20]. Для увеличения скорости расчета хеш идентификаторов необходимо разработать специализированное устройство, повышающее быстродействие данного процесса, что позволит использовать предложенные методы обезличивания данных при организации высокоскоростного обмена сообщениями с требуемым уровнем конфиденциальности.

Разработка структурно-функциональной схемы устройства формирования уникальной последовательности

На основе предложенной математической модели обезличивания данных разработано устройство формирования уникальной последовательности, которое работает следующим образом.

Путём конкатенации из выбранного перечня атрибутов формируется единая строка. Данная строка преобразуется в битовую последовательность. Полученная битовая последовательность дополняется до количества кратному 576 бит. Дополнение происходит по следующему правилу, где *M*- исходное сообщение:

$$M=M||0x01||0x00||..||0x00||0x80$$

К сообщению дописывается единичный байт 0x01, необходимое количество нулей, ко всей последовательности необходимо добавить байт со значением 0x80.Если же необходимо дополнить всего один байт, то достаточно добавить байт 0x81.

На следующем этапе происходит инициализация исходного состояния - формируется массив размерностью 5x5, элементами которого являются 64-битные слова, изначально все нули. Далее проводится сложение по модулю 2 фрагментов исходного сообщение с фрагментами исходного состояния по следующей формуле:

$$S[x,y] := S[x,y] \oplus Pi[x+5y] \text{ при } x= 0..5 , y= 0..5; \quad (5),$$

где знаком ":=" обозначается оператор присваивания. Знаком "⊕" обозначена побитовая операция суммирования по модулю 2.

На следующем этапе происходит псевдослучайная перестановка внутреннего состояния. 24 раза с внутренним состоянием S происходят преобразования в соответствии (1) - (4).

Массив *A* представляет собой предопределенный набор значений, в котором указывается, на сколько бит необходимо производить циклический сдвиг.

Значения всех элементов массива *A* представлены в таблице 1.

Массив RC представляет из себя набор констант, которые являются предопределенными для каждого *i*-го раунда. Значения этих констант указаны в таблице 2.

Таблица 1 - Значения элементов массива A

	x=3	x=4	x=0	x=1	x=2
y=2	25	39	3	10	43
y=1	55	20	36	44	6
y=0	28	27	0	1	62
y=4	56	14	18	2	61
y=3	21	8	41	45	15

Таблица 2 - Значения констант массива RC

Элемент массива	Значение	Элемент массива	Значение
RC[0]	0x0000000000000001	RC[12]	0x000000008000808B
RC[1]	0x0000000000008082	RC[13]	0x800000000000008B
RC[2]	0x800000000000808A	RC[14]	0x8000000000008089
RC[3]	0x8000000080008000	RC[15]	0x8000000000008003
RC[4]	0x000000000000808B	RC[16]	0x8000000000008002
RC[5]	0x0000000080000001	RC[17]	0x8000000000000080
RC[6]	0x8000000080008081	RC[18]	0x000000000000800A
RC[7]	0x8000000000008009	RC[19]	0x800000008000000A
RC[8]	0x000000000000008A	RC[20]	0x8000000080008081
RC[9]	0x0000000000000088	RC[21]	0x8000000000008080
RC[10]	0x0000000080008009	RC[22]	0x0000000080000001
RC[11]	0x000000008000000A	RC[23]	0x8000000080008008

Результирующее значение получается из повторных преобразований по формулам предыдущего этапа для всех M_i блоков, и вывода первых 512 бит сообщения. После указанных преобразований уникальная последовательность, получившаяся из преобразования критически важных атрибутов, становится идентификатором субъекта, что позволяет исключить из обработки часть персональных данных и перевести их в обезличенную форму.

Структурная схема устройства представлена на рисунке 2 и содержит генератор тактовых сигналов 1, вход которого подключён к питанию (+5V) 9, а выход подключен ко всем элементам устройства, блок дополнения 2, блок генерации внутреннего состояния 3, блок синхронизации 4, блок побитовой операции сумма по модулю два с 64-битными словами 5, блоки псевдослучайных перестановок 6, 7 и счётчик прямоугольных импульсов 8, подключается USB-интерфейс, в котором каналы Data+ и Data- являются информационными, а +5V и GRD используются для питания устройства.

Устройство работает следующим образом.

Для того, чтобы устройство начало принимать на вход информационные сигналы, на вход in_ready должен быть подан сигнал 1, а на входы reset 0. Только в этом случает устройство, будет принимать информационные входные сигналы. Входной сигнал in_ready становится равен 1, сразу после подключения устройства к ЭВМ. Изначально reset=0.

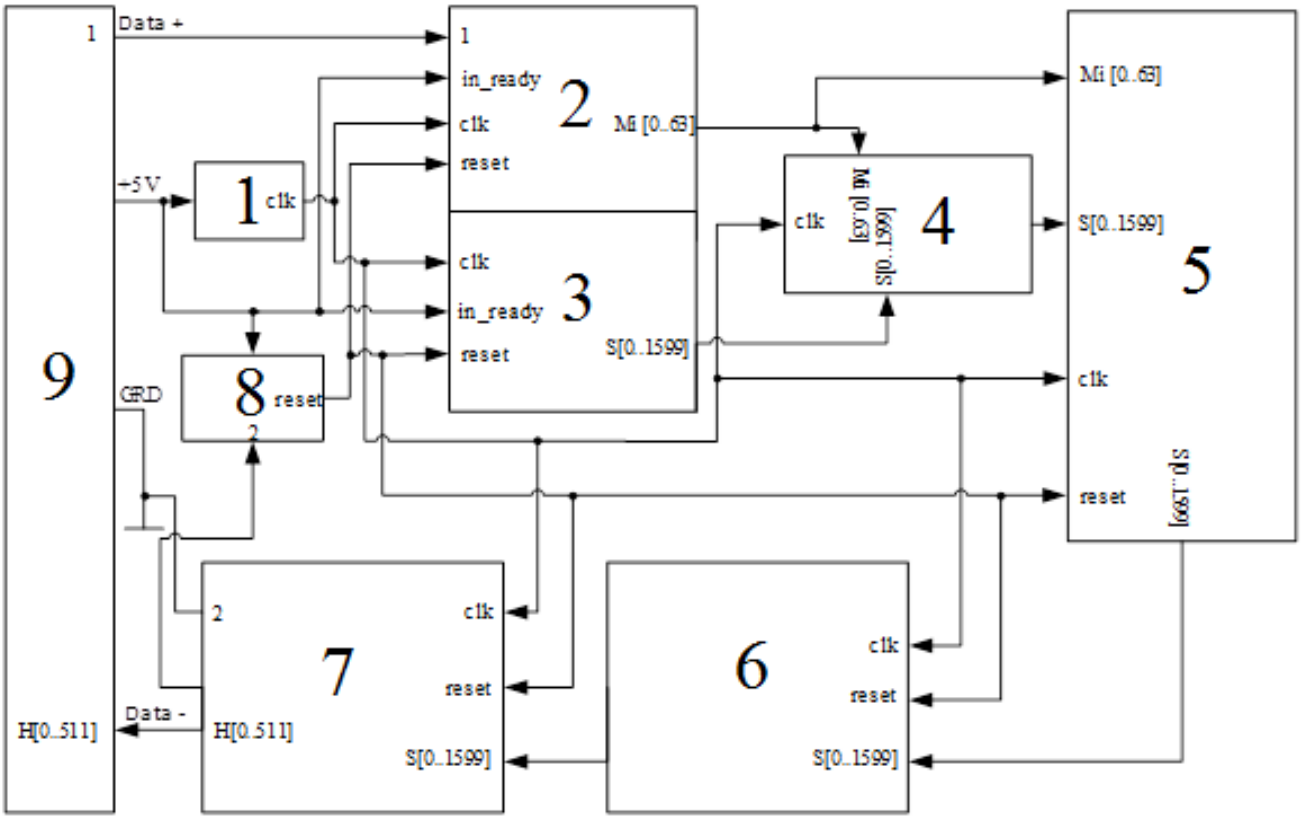

Рисунок 2 - Устройство формирования уникальной последовательности, используемой при обезличивании персональных данных

В блоке дополнения 2 битовая последовательность, поступившая по каналу 1 из ЭВМ, дополняется до длины кратной 576, и выводится 64-битными последовательностями. В блоке генерации внутреннего состояния 3 инициализируется внутренне состояние. Оно представляет из себя массив 1600 бит, все элементы которого нули. В блоке синхронизации 4 происходит синхронизация поступающих фрагментов исходных данных и внутреннего состояния по тактовой частоте. В блоке побитовой операции сумма по модулю два с 64-битными словами 5, с внутренним

состоянием условно поделенным на двумерный массив величиной 5x5 64-битных слов по формуле (5) фрагменты исходного сообщение суммируются по модулю 2 с фрагментами исходного состояния. Выходными данными является внутреннее состояние 1600 бит. В блоках псевдослучайных перестановок 6,7 перемешиваются биты внутреннего состояния по формулам (1)-(4).

Сущность перестановок поясняется рисунками 3,4, где пронумерованными квадратами обозначены 64 битные слова. Если эти слова представить двумерным массивом, то нумерация будет выглядеть как в таблице 3:

Таблица 3 - Внутреннее состояние матрицы, элементы которой – 64-битные слова

04	14	24	34	44
03	13	23	33	43
02	12	22	32	42
01	11	21	31	41
00	10	20	30	40

Выходными данными будут является первые 512 бит. Они поступают на канал Data- и отправляются в плагин, который установит полученную последовательность идентификатором субъекта информационной системы.

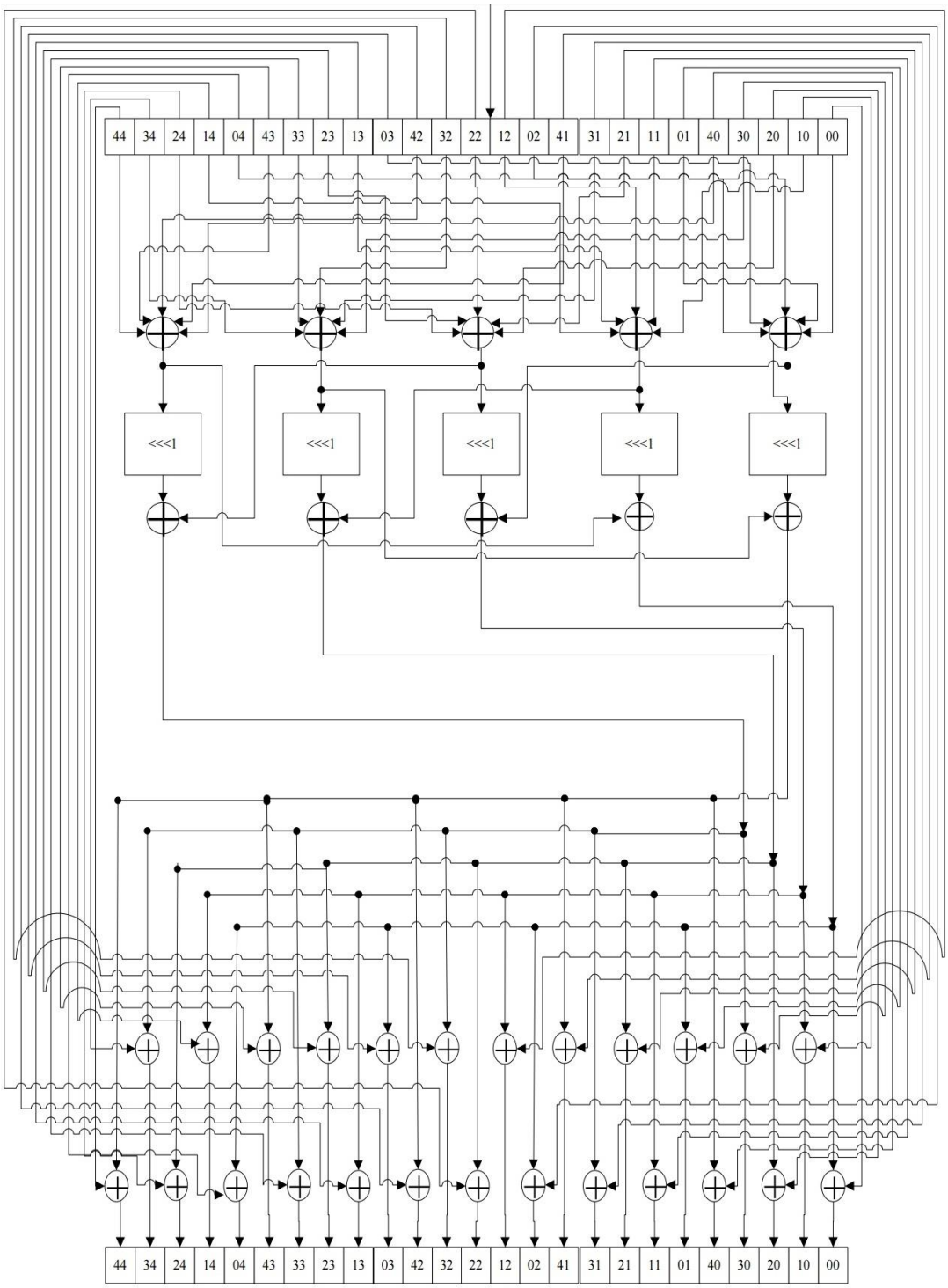

Рисунок 3 – Схема перестановок

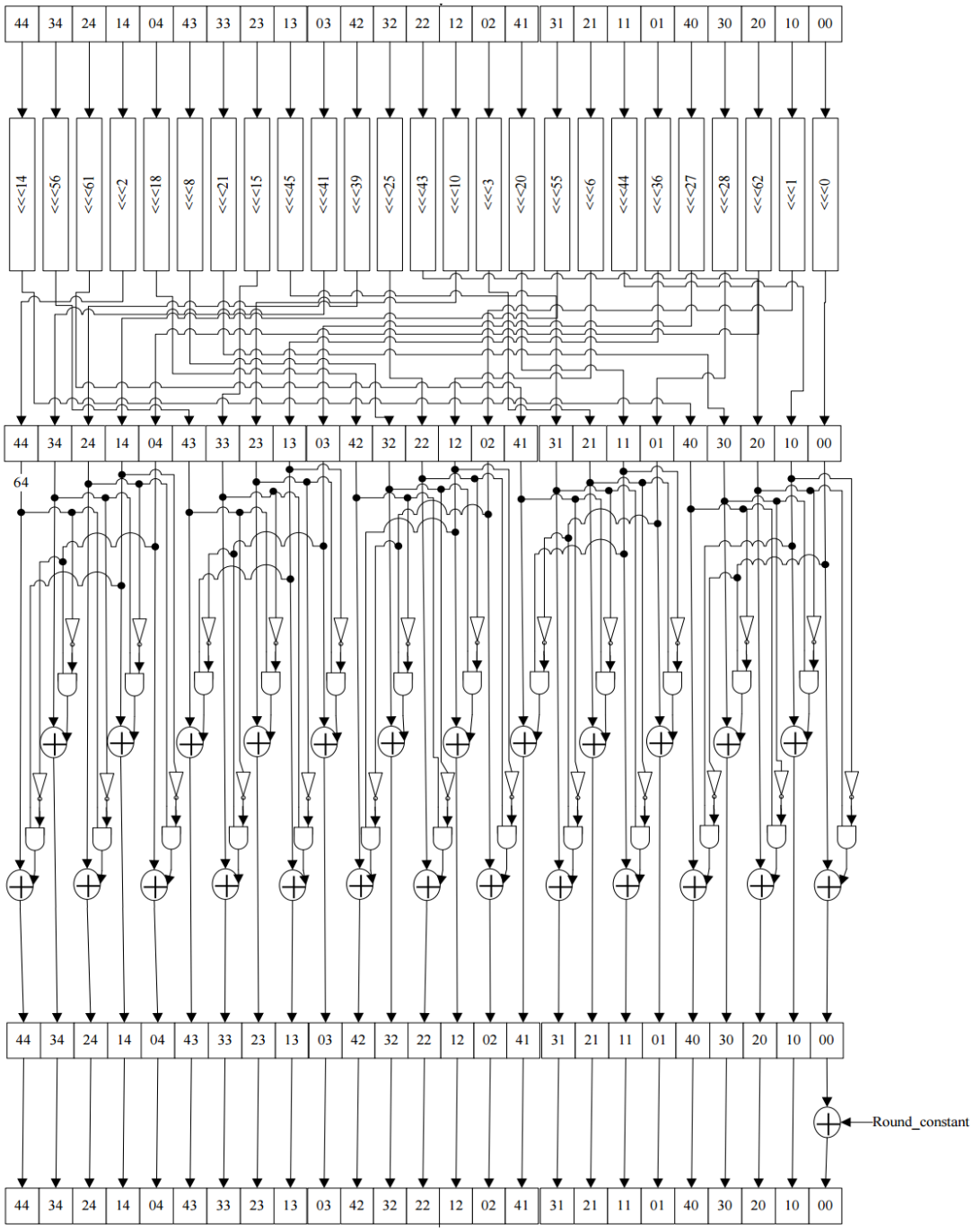

Рисунок 4 – Схема перестановок

В то же время, когда счётчик прямоугольных импульсов 8 подсчитает 512 прямоугольных сигналов на выход reset будет подан сигнал 1, что не позволит устройству принимать на вход информационные сигналы. Если на входе ожидается

новые исходные данные следующего субъекта, то после одного цикла reset снова станет равным 0, и преобразование данных продолжится.

Результаты экспериментальных исследований

Для проведения экспериментов была смоделирована ситуация по обезличиванию данных пассажиров авиакомпании, произведена выборка данных одного миллиона записей.

На основе исходной выборки были определены ключевые критически важные атрибуты, однозначно идентифицирующие субъекта персональных данных, которые хранятся в защищенной информационной системе. Используя эти данные и приватный ключ, для каждой записи формируется хэш идентификатор, который является первичным ключом субъекта персональных данных в обезличенной информационной системе. Пример исходной выборки представлен в таблице 4.

Таблица 4 – Пример исходной выборки

Фамилия	Имя	Пол	Дата рождения	Паспорт	Рейс	Дата, время	Место
Svetlova	Svetlana	MRS	12.12.1972	3804 705689	SU1512	20.11.2020/ 01:35	23A
Temnova	Olga	MRS	11.11.1990	3812 558964	SU2550	21.11.2020/ 23:50	15D

Хеш идентификатор формировали при помощи разработанного программного средства, работающего на ПК с процессором Intel Xeon E5-2650 и при помощи разработанного устройства. Сравнительные характеристики представлены в таблице 5.

Таблица 5 – Сравнительные характеристики устройств

Показатель	Intel Xeon E5-2650	Разработанное устройство
Скорость, Кхеш/с	129.4	163581.3
Потребляемая мощность устройства, Вт	350	30
Стоимость, руб	60000	5000
Вес, кг	6	0,3

Пример результирующей выборки представлен в таблице 6.

Таблица 6 – Пример результирующей выборки

Хеш идентификатор	Рейс	Дата, время	Место
1628g3db5g13865ada5856a630a736653059fc7e2d7c49f897b636428c62a26b	SU1512	20.11.2020/ 01:35	23A
4d954s630cfaafe3dd151a2e06d7345a44a61877a8c097622abfd6ca0f515a7f	SU2550	21.11.2020/ 23:50	15D

Выводы

Решена актуальная задача повышения быстродействия при обезличивании данных в информационной системе методом введения идентификаторов с использованием хеширования критически важных данных и приватного ключа, путем внедрения разработанного устройства.

Научная новизна полученных результатов заключается в том, что предложен метод введения идентификаторов с использованием хеширования критически важных данных и приватного ключа. Что позволяет повысить уровень конфиденциальности

данных, снизить требования к уровню защищённости информационной системы, повысить скорость обработки данных за счет свертки критически важных данных в хеш идентификатор с использованием параллельных вычислений с использованием разработанного специализированного устройства.

Практическая значимость полученных результатов заключается в том, что разработано устройство формирования уникальной последовательности, используемой при обезличивании персональных данных, реализующее предложенный метод, а также проведены эксперименты по подтверждению адекватности предложенной математической модели. Результаты эксперимента позволяют рекомендовать предложенное устройство для внедрения в автоматизированные информационные системы обработки персональных данных на этапе проектирования или оптимизации, а также в системы обмена данными по радиоканалам, когда требуется соблюсти требуемый уровень конфиденциальности и обеспечить адресность данных. Так при организации спутниковой широковещательной передаче данных, абоненты могут получать адресный контент, в соответствии с их хешидентификатором. В авиационной и космической отрасли использование разработанного устройства позволит повысить защищенность данных при функционировании различных систем мониторинга.

Библиографический список

1. Родичев Ю.А. Нормативная база и стандарты в области информационной безопасности. – СПб.: Издательский дом «Питер», 2018. – 255 р.

2. Сычев Ю.В. Стандарты информационной безопасности. Защита и обработка конфиденциальных документов. – Саратов: Вузовское образование, 2019. – 223 p.

3. Бондаренко К.О., Козлов В.А. Универсальный быстродействующий алгоритм процедур обезличивания данных // Известия ЮФУ. Технические науки. 2015. № 11 (172). С. 130 – 142.

4. Трифонова Ю.В., Жаринов Р.Ф. Возможности обезличивания персональных данных в системах, использующих реляционные базы данных // Доклады ТУСУР. 2014. № 2 (32). С. 188 – 194.

5. Ажмухамедов И.М., Демина Р.Ю., Сафаров И.В. Системный подход к обеспечению конфиденциальности обезличенных персональных данных в учреждениях здравоохранения // Современные проблемы науки и образования. 2015. № 1-1. URL: https://science-education.ru/ru/article/view?id=18610

6. Елисеев С.О., Крюков Д.А. Система криптографической генерации идентичных данных на основе алгоритма Диффи-Хеллмана // Труды МАИ. 2018. № 101. URL: http://trudymai.ru/published.php?ID=97041

7. Глебов О.И. Специализированная система электронного документооборота // Труды МАИ. 2005. № 18. URL: http://trudymai.ru/published.php?ID=34190

8. Bertoni G., Daemen J., Peeters M., Van G. Keccak code package. URL: https://github.com/gvanas/KeccakCodePackage

9. Huang S., Xu G., Wang M. et al. Conditional cube attack on reduced-round Keccak sponge function // Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, 2017, pp. 259 - 288. DOI: 10.1007/978-3-319-56614-6_9

10. Guo J., Liu M., Song L. Linear structures: Applications to cryptanalysis of round-reduced Keccak // International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, 2016, pp. 249 – 274. DOI: 10.1007/978-3-662-53887-6_9

11. Jeethu J., Karthika R., Nandakumar R. Design and characterization of SHA 3- 256 Bit IP core // International conference on emerging trends in engineering, science and technology, 2016, vol. 24, pp. 918 – 924. DOI: 10.1016/j.protcy.2016.05.184

12. Dinur I., Morawiecki P., Pieprzyk J. et al. Cube attacks and cube-attack-like cryptanalysis on the round-reduced Keccak sponge function // Eurocrypt: Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, 2015, pp. 733 – 761. 10.1007/978-3-662-46800-5_28

13. Spevakov A.G., Spevakova S.V., Primenko D.V. Method of data depersonalization in protected automated information systems // Radio Electronics, Computer Science, Control, 2020, no. 1, pp. 162 - 168. DOI: 10.15588/1607-3274-2020-1-16

14. Primenko D.V., Spevakov A.G., Spevakova S.V. Depersonalization of Personal Data in Information Systems // International Russian Automation Conference, Springer, Cham, 2019, pp. 763 - 770. DOI: 10.1007/978-3-030-39225-3_83

15. Ноздрина А.А., Спеваков А.Г., Применко Д.В. Способ деперсонализации персональных данных. Патент РФ 2636106, МПК G06F 12/14, G06F 12/14. Бюл. № 32, 04.07.2016.

16. Таныгин М.О. Восстановление порядка следования информационных пакетов на основе анализа хеш-последовательностей // Известия Юго-Западного государственного университета. 2020. Т. 24. № 1. С. 175 - 188. DOI: 10.21869/2223-1560-2020-24-1-175-188

17. Таныгин М.О. Алгоритм определения источника фрагментированных сообщений // Известия высших учебных заведений. Приборостроение. 2020. Т. 63. № 8. С. 702 - 710. DOI: 10.17586/0021-3454-2020-63-8-702-710

18. Хмара С.А., Назаров А.В. Минимизация оборудования устройства управления цифровым вычислительным устройством // Радиотехника. 2011. № 49. С. 59.

19. Симонов А.С., Семенов А.С., Макагон Д.В. Направления развития высокоскоростной коммуникационной сети для многопроцессорных вычислительных систем // Труды МАИ. 2019. № 108. URL: http://trudymai.ru/published.php?ID=109525

20. Матафонов Д.Е. Создание и отработка маршрутизатора в стандарте SpaceWire на отечественной программируемой логической интегральной схеме // Труды МАИ. 2018. № 103. URL: http://trudymai.ru/published.php?ID=100780

A unit for unique sequence generating employed while personal data impersonalizing

Spevakov A.G.*, Kaluckiy I,V.**

South-Western State University,

94, 50-let Oktyabrya str., Kursk, 305040, Russia

**e-mail: aspev@yandex.ru*

***e-mail: kaluckiy_igor@yandex.ru*

Abstract

A problem of data impersonalizing occupies special place in automated information systems while data processing. Such methods and algorithms are employed to improve the system security and reduce material costs for information security tools purchasing. One of the tasks consists in developing specialized units that will allow increasing the speed of data conversion during impersonalizing. Such units can be employed not only in large data processing centers, but also in information systems with a small number of personal data subjects, but demanding data processing speed. This allows performance improvement, as well operating costs and product weight reduction. The purpose of this work consist in the speed increase of the unit for generating a unique sequence for personal data impersonalizing by parallel computing and FPGA based implementation. The problem solution of increasing the speed of forming a unique sequence for impersonalizing personal data is achieved by implementing a method and algorithm for data impersonalizing that allows their FPGA based implementation employing parallel computing. The developed unit novelty lies in the fact that a method for introducing identifiers using hashing critical data and a private key was

proposed. This allows increasing the of data privacy level, bating requirements for the security level of the information system, and increasing the of data processing speed by convolving critical data into a hash identifier by parallel computing employing the developed specialized device. The results obtained consist in the development of a unit for generating a unique sequence used for personal data impersonalizing. This unit implements the proposed method. The experiments confirming the proposed mathematical model adequacy in comparison with the software implementation were conducted. The proposed device allowed increasing the speed of hash identifiers computing by more than a thousand times, with power consumption decrease by 11.7 times. The results of the experiment allow recommending the proposed unit for implementation in automated information systems for personal data processing at the design stage or optimization of existing systems. It will reduce the information protecting system cost. The authors propose the presented solution implementation in the form of a FPGA based unit, which will allow increasing speed and reducing operation costs.

Keywords: data impersonalizing, hashing, conversion, high-speed unit.

References

1. Rodichev Yu.A. *Normativnaya baza i standarty v oblasti informatsionnoi bezopasnosti* (Normative base and standards in the field of information security), Saint Petersburg, Izdatel'skii dom “Piter”, 2018, 255 p.

2. Sychev Yu.V. *Standarty informatsionnoi bezopasnosti. Zashchita i obrabotka konfidentsial'nykh dokumentov* (Standards of information security. Protection and processing of confidential documents), Saratov, Vuzovskoe obrazovanie, 2019, 223 p.

3. Bondarenko K.O., Kozlov V.A. *Izvestiya YuFU. Tekhnicheskie nauki*, 2015, no. 11 (172), pp. 130 - 142.

4. Trifonova Yu.V., Zharinov R.F. *Doklady TUSUR*, 2014, no. 2 (32), pp. 188 – 194.

5. Azhmukhamedov I.M., Demina R.Yu., Safarov I.V. *Sovremennye problemy nauki i obrazovaniya*, 2015, no. 1-1. URL: https://science-education.ru/ru/article/view?id=18610

6. Eliseev S.O., Kryukov D.A. *Trudy MAI*, 2018, no. 101. URL: http://trudymai.ru/eng/published.php?ID=97041

7. Glebov O.I. *Trudy MAI*, 2005, no. 18. URL: http://trudymai.ru/eng/published.php?ID=34190

8. Bertoni G., Daemen J., Peeters M., Van G. *Keccak code package*. URL: https://github.com/gvanas/KeccakCodePackageeng/

9. Huang S., Xu G., Wang M. et al. Conditional cube attack on reduced-round Keccak sponge function, *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Paris, France, 2017, pp. 259 - 288. DOI: 10.1007/978-3-319-56614-6_9

10. Guo J., Liu M., Song L. Linear structures: Applications to cryptanalysis of round-reduced Keccak, *International Conference on the Theory and Application of Cryptology and Information Security*, Hanoi, Vietnam, 2016, pp. 249 – 274. DOI: 10.1007/978-3-662-53887-6_9

11 .Jeethu J., Karthika R., Nandakumar R. Design and characterization of SHA 3- 256 Bit IP core, *International conference on emerging trends in engineering, science and technology*, 2016, vol. 24, pp. 918 – 924. DOI: 10.1016/j.protcy.2016.05.184

12. Dinur I., Morawiecki P., Pieprzyk J. et al. Cube attacks and cube-attack-like cryptanalysis on the round-reduced Keccak sponge function, *Eurocrypt: Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Sofia, Bulgaria, 2015, pp. 733 – 761. DOI: 10.1007/978-3-662-46800-5_28

13. Spevakov A.G., Spevakova S.V., Primenko D.V. Method of data depersonalization in protected automated information systems, *Radio Electronics, Computer Science, Control*, 2020, no. 1, pp. 162 - 168. DOI: 10.15588/1607-3274-2020-1-16

14. Primenko D.V., Spevakov A.G., Spevakova S.V. Depersonalization of Personal Data in Information Systems, *International Russian Automation Conference*, Springer, Cham, 2019, pp. 763 - 770. DOI: 10.1007/978-3-030-39225-3_83

15. Nozdrina A.A., Spevakov A.G., Primenko D.V. *Patent RF 2636106*, MPK G06F 12/14, G06F 12/14, 04.07.2016.

16. Tanygin M.O. *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta*, 2020, vol. 24, no. 1, pp. 175 - 188. DOI: 10.21869/2223-1560-2020-24-1-175-188

17. Tanygin M.O. *Izvestiya vysshikh uchebnykh zavedenii. Priborostroenie*, 2020, vol. 63, no. 8, pp. 702 - 710. DOI: 10.17586/0021-3454-2020-63-8-702-710

18. Khmara S.A., Nazarov A.V. *Radiotekhnika*, 2011, no. 49, pp. 59.

19. Simonov A.S., Semenov A.S., Makagon D.V. *Trudy MAI*, 2019, no. 108. URL: http://trudymai.ru/eng/published.php?ID=109525

20. Matafonov D.E. *Trudy MAI*, 2018, no. 103. URL: http://trudymai.ru/eng/published.php?ID=100780