Раздел I. Алгоритмы обработки информации

УДК 004.021 DOI 10.18522/2311-3103-2021-2-6-18

Л.К. Бабенко, Е.А. Толоманенко

ГИБРИДНОЕ ШИФРОВАНИЕ НА ОСНОВЕ ИСПОЛЬЗОВАНИЯ СИММЕТРИЧНЫХ И ГОМОМОРФНЫХ ШИФРОВ*

Целью данной работы является разработка и исследование гибридного алгоритма шифрования на основе совместного применения симметричного алгоритма шифрования Кузнечик и гомоморфного шифрования (Схемы Джентри или схемы BGV). Такой алгоритм шифрования может быть востребован в ситуациях ограниченных вычислительных ресурсов. Смысл заключается в том, что при правильном выражении основных операций симметричного алгоритма шифрования через булевы функции, появляется возможность на передающей стороне зашифровать данные симметричным шифром, а секретный ключ шифрования – гомоморфным. В таком случае на стороне приема можно провести манипуляции так, чтобы исходное зашифрованное сообщение оказалось также зашифровано только гомоморфным шифром. При этом симметричное шифрование снимается, но информация остается недоступной обрабатывающему ее узлу. Такое свойство секретности позволяет проводить ресурсоемкие операции на мощном вычислительном узле, предоставляя гомоморфно зашифрованные данные для малоресурсного узла с целью их последующей обработки в зашифрованном виде. В статье представлен разработанный гибридный алгоритм. В качестве симметричного алгоритма шифрования использован алгоритм шифрования Кузнечик, являющийся частью стандарта ГОСТ Р34.12 – 2015. Для того, чтобы иметь возможность применять гомоморфное шифрование к данным, зашифрованным шифром Кузнечик, S-блок замены алгоритма Кузнечик представлен в булевом виде с использованием полинома Жегалкина. Также линейное преобразование L представлено в виде последовательности выполнения простейших операций сложения и умножения над преобразуемыми данными. Первичное моделирование разрабатываемого алгоритма было проведено на упрощенной версии алгоритма Кузнечик S-KN1.

Криптография; блочный шифр; симметричный шифр; алгоритм Кузнечик; гомоморфное шифрование; гибридное шифрование; полином Жегалкина; булева функция.

L.K. Babenko, E.A. Tolomanenko

HYBRID ENCRYPTION BASED ON SYMMETRIC AND HOMOMORPHIC CIPHERS

The purpose of this work is to develop and research a hybrid encryption algorithm based on the joint application of the symmetric encryption algorithm Kuznyechik and homomorphic encryption (Gentry scheme or BGV scheme). Such an encryption algorithm can be useful in situations with limited computing resources. The point is that with the correct expression of the basic operations of the symmetric encryption algorithm through Boolean functions, it becomes possible on the transmitting side to encrypt the data with a symmetric cipher, and the secret encryption key - with a homomorphic one. In this case, manipulations can be carried out on the receiving side so that the original encrypted message is also encrypted only with a homomorphic cipher. In this case, symmetric encryption is removed, but the information remains inaccessible to the node that processes it. This property of secrecy makes it possible to carry out resource-intensive operations on

* Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта РФФИ №19-37-90148.

a powerful computing node, providing homomorphically encrypted data for a low-resource node for the purpose of their subsequent processing in encrypted form. The article presents the developed hybrid algorithm. As a symmetric encryption algorithm, Kuznyechik encryption algorithm is used, which is part of the GOST R34.12 - 2015 standard. In order to be able to apply homomorphic encryption to data encrypted with the Kuznyechik cipher, the Kuznyechik algorithm S-boxes is presented in a boolean form using the Zhegalkin polynomial. Also, the linear transformation L is presented in the sequence form of performing the simplest operations of addition and multiplication on the transformeddata. The primary modeling of the developed algorithm was carried out on a simplified version of the KuzchyechikS-KN1 algorithm.

Cryptography; block cipher; symmetric cipher; Kuznyechik cipher; homomorphic encryption; hybrid encryption; Zhegalkin polynomial; Boolean function.

Введение. В последнее десятилетие стремительно развиваются информационные технологии. Активно развивается отрасль интернета вещей (IoT), робототехнические системы, сенсорные сети. Так как подобные устройства, как правило обладают ограниченным вычислительным ресурсом, то в первую очередь при разработке таких средств внимание уделяется вопросам безопасности, и в частности вопросам шифрования данных. Об актуальности проблем, связанных с обеспечением криптографической надежности, говорит, например, ряд докладов, представленных на ежегодной научной конференции «РусКрипто-2021» [1–3].

Область применения подобных устройств разнообразна, затрагивает различные сферы деятельности человека. Это и контроль экологических параметров окружающей среды, и контроль медицинских показателей пациентов, и контроль доступа к различным удаленным системам мониторинга, геологоразведочные работы, и многое другое. Часто подобные устройства обрабатывают и передают информацию, которая может носить конфиденциальный характер (например, медицинские показатели пациента, которые составляют врачебную тайну) или информацию, передача которой в открытом виде может быть скомпрометирована или изменена с целью кибератаки (например, при мониторинге параметров объекта в режиме реального времени). Это явным образом показывает необходимость разработки и внедрения алгоритмов и протоколов, направленных на защиту передаваемых и обрабатываемых данных.

Частично данный вопрос решается с использованием легковесной криптографии [4]. Тем не менее, легковесная криптография не позволяет обрабатывать данные в зашифрованном виде. Это означает, что устройство должно иметь механизмы для зашифрования и расшифрования информации с целью ее обработки. Кроме того, секретный ключ шифрования должен быть у всех узлов, участвующих в обработке информации. Это в свою очередь накладывает необходимость применения стойких, но при этом ресурсонезатратных протоколов формирования и распределения ключа.

Одним из вариантов решения сложившейся проблемы казалось появление нового вида шифров, названных гомоморфными [5–9]. Впервые гомоморфная схема шифрования была предложена в 2009 году сотрудником IBM Крейгом Джентри. Данная схема представляет собой асимметричную полностью гомоморфную схему шифрования на основе идеальных решеток [10]. В схеме Джентри имелся явный недостаток. При выполнении многократных действий в данных накапливалась ошибка. При превышении определенного порогового значения данные становились недешифруемыми. В 2012 году Джентри показал, как можно выполнять самокоррекцию шифров [11] для того, чтобы данные по-прежнему могли подвергаться обратному преобразованию. Гомоморфные шифры позволяют обрабатывать информацию в зашифрованном виде. В частично гомоморфной схеме доступен только один вид операций для обработки. Как правило, это сложение или умножение. В полностью гомоморфной схеме доступны обе операции для выполнения преобразований [12]. Однако у гомоморфных шифров есть ряд особенностей, накладывающих ограничение на их применение. И самая главная особенность – это большой объем памяти, который требуется для выполнения гомоморфных опера-

ций. Данный факт напрямую связан с ограничением применения гомоморфных шифров для защиты информации в малоресурсной отрасли [13, 14]. Решением сложившейся проблемы может являться использование гибридных схем шифрования, сочетающих в себе использование симметричных и гомоморфных шифров. В 2012 году Джентри показал, как можно использовать схему полностью гомоморфного шифрования совместно с алгоритмом шифрования AES [15].

В настоящей статье предлагается рассмотреть возможность создания гибридной схемы шифрования на основе использования отечественного стандарта шифрования данных ГОСТ Р 34.12-2015 и схемы гомоморфного шифрования BGV.

Статья организована следующим образом. В первой главе приводится краткое описание тех алгоритмов шифрования, которые предполагается использовать при создании гибридной схемы. Во второй главе описывается общая формулировка протокола обмена данными между двумя узлами сети при использовании разрабатываемой гибридной схемы шифрования. В третьей главе представлена разработка алгоритмов для представления базовых алгоритмов шифрования Кузнечик в виде конъюнктивной нормальной формы (КНФ). В главе 4 представлен разработанный алгоритм для гибридной схемы.

Постановка задачи. Разработать гибридный алгоритм шифрования на основе использования симметричного и гомоморфного видов шифров для возможности осуществления операций в малоресурсных системах. В качестве симметричного шифра предполагается использовать алгоритм шифрования Кузнечик, входящий в состав отечественного стандарта шифрования данных ГОСТ Р 34.12-2015, а в качестве алгоритма гомоморфного шифрования схему BGV.

1. Краткое описание используемых алгоритмов шифрования

1.1. Алгоритм шифрования Кузнечик. Алгоритм шифрования Кузнечик представляет собой симметричный блочный шифр со следующими параметрами: длина блока – 128 бит, длина ключа– 256 бит, количество раундов шифрования – 9, схема построения – SP-сеть. Каждый раунд шифрования содержит три базовые операции: сложение данных с раундовым ключом посредством операции xor, замена байтов с помощью S-блока замены (операция S) и линейное преобразование (операция L). Схематичное описание одного раунда шифрования для алгоритма Кузнечик приведено на рис. 1.

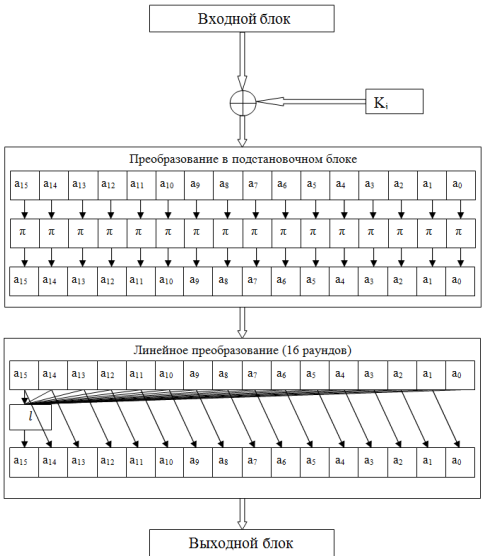

Рис. 1. Один раунд шифрования

Для SP-сети всегда требуется раундовых ключей на 1 больше, чем используемое количество раундов шифрования, поэтому всего необходимо 10 раундовых ключей. Они вырабатываются из исходного 256-битного мастер-ключа с применением схемы Фейстеля как показано на рис. 2. В каждом раунде используется функция F, состоящая из трех операций: xor преобразуемых данных с раундовой константой, преобразование с помощью S-блока замены и линейное преобразование L, которые изображены на рис. 3. С подробным описанием работы алгоритма Кузнечик можно познакомиться в [16, 17].

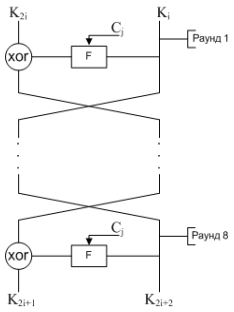

Рис. 2. Сеть Фейстеля для выработки раундовых ключей

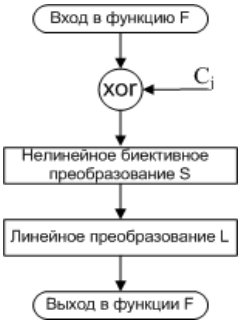

Рис. 3. Функция F, используемая в сети Фейстеля

1.2. Упрощенный алгоритм шифрования Кузнечик S-KN1. Упрощенный алгоритм Кузнечик (S-KN1) был разработан коллективом кафедры БИТ ИКТИБ ЮФУ и впервые представлен на международной конференции SIN-2017 [18]. Схема S-KN1 полностью повторяет структуру оригинального шифра Кузнечик и представлена на рис. 4.

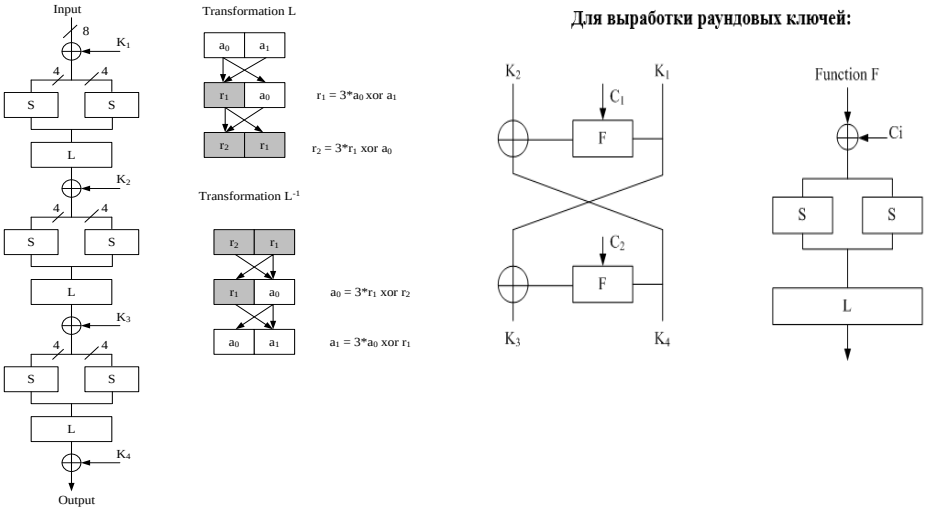

Рис. 4. Схема упрощенного алгоритма S-KN1

S-KN1 представляет собой симметричный блочный шифр, основанный на SP-сети и преобразовании информации в трех раундах. Длина преобразованного блока 8 бит; длина секретного ключа составляет 16 бит. Алгоритм начинает работу со сложения исходных данных с подключом K1. За этой операцией следуют три раунда шифрования. В каждом раунде сообщение разбивается на два полубайта, ка-

ждый из которых проходит через блок замены S, затем части объединяются и проходят через линейное преобразование L. Каждый раунд заканчивается сложением с соответствующим раундовым подключом.

1.3. Схема BGV(Brakerski-Gentry-Vaikuntanathan). Схема BGV была предложена тройкой ученых, в состав которых входит автор первого алгоритма гомоморфного шифрования Крейг Джентри [19]. Схема BGV впервые предложила решить проблему недешифруемости текстов без использования самокоррекции, вместо этого используются две другие операции: смена ключа и смена модуля. В схеме BGV множество открытых текстов представляется в виде кольца Rp, а множество зашифрованных текстов – в виде кольца Rq. При этом числа p и q являются простыми и связаны соотношением: $q \equiv 1 \bmod p$. Для реализации схемы BGV используется пять базовых операций.

1. Операция генерации ключа KeyGen
   s',e ← χ;
   a ← Rq;
   b ← −(as' + pe);
   sk ← s = (1, s') // приватный ключ
   pk ← k = (b,a) // публичный ключ

2. Операция зашифрования Enc (выполняется с использованием публичного ключа получателя)
   r ← Rp // случайное число
   e ← χ2 // распределение хи-квадрат
   m ← (m,0) // шифруемое значение
   c ← m + kr + pe // результат шифрования

3. Операция расшифрования Dec (выполняется с использованием приватного ключа получателя)
   m ← $[[\langle c|s\rangle]_q]_p$ // скалярное произведение шифрованного сообщения и приватного ключа берется сначала по модулю q, а затем берется по модулю p.

4. Операция смены ключа KeySwitch. Применяется для коррекции чрезмерного роста длин ключей: $[[\langle c|s\rangle]_q]_p = [[\langle c'|s'\rangle]_q]_p$

5. Операция смены модуля ModulusSwitch. Применяется для уменьшения зашумленности шифртекста. Заменяет пару c, q на пару c', q' так, что: $[[\langle c|s\rangle]_q]_p = [[\langle c'|s\rangle]_{q'}]_p$

При этом для двух текстов m_1, m_2 зашифрованных как c_1, c_2 выполняются соотношения: $m_1 + m_2 = [[\langle c_1 + c_2|s\rangle]_q]_p$ и $m_1 \otimes m_2 = [[c_1 \otimes c_2|s \otimes s]_q]_p$.

Таким образом, сложению текстов соответствует сложение шифртекстов, а умножению текстов — тензорное произведение шифртектов и изменение секретного ключа s на $s \otimes s$. Таким образом, в процессе гомоморфного шифрования на каждом уровне по умножению производится смена как модулей, так и ключей.

2. Протокол обмена данными между двумя узлами сети при использовании разрабатываемой гибридной схемы шифрования. На основании идеи Джентри, а также с использованием особенностей сенсорных сетей, был разработан протокол обмена данными между двумя узлами сети при использовании разрабатываемой гибридной схемы шифрования. Разработанный протокол предназначен для задач сети, в которой есть узел, обладающий потенциально мощной вычислительной мощностью (База) и узлы с небольшим вычислительным ресурсом (например, сенсоры). Пример такой схемы представлен на рис. 5.

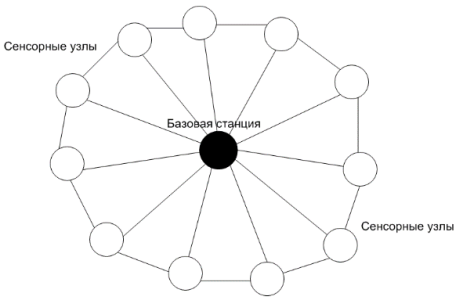

Рис. 5. Пример конфигурации сети, предназначенной для использования разработанного протокола

Важно отметить, что предложенный протокол будет практичным только в том случае, если ни база, ни сенсоры не владеют информацией о секретном ключе. То есть в данном случае база выступает в качестве мощного вычислительного ресурса. Она производит вычисления над данными, не зная самих данных (по аналогии с облачными вычислениями). Например, такие сенсоры нужны для секретной разведки. Тогда все базы оперируют открытыми ключами, не зная саму информацию. А вскрыть данные может только доверенное лицо, у которого есть секретный ключ. В противном случае теряется весь смысл разработанного протокола.

Протокол 1:

На стороне **Сенсорного узла:**

1. Определяются стартовые данные Data
2. Генерируется секретный ключ K
3. Данные шифруются с использованием алгоритма Кузнечик: EncKuzn(Data, K)
4. Ключ K шифруется с использованием полностью гомоморфного шифрования Enc_FHE(K)
5. От сенсорного узла к базовой станции передаются значения EncKuzn(Data, K) и Enc_FHE(K)

На стороне **Базовой станции**:

1. Каждый бит зашифрованного сообщения EncKuzn(Data, K) шифруется с использованием полностью гомоморфного шифрования Enc_FHE(EncKuzn(Data, K))
2. Выполняются функции расшифрования симметричного шифра с использованием гомоморфных операций Enc_FHE(EncKuzn(Data, K)) = Enc_FHE(Data). Данные остаются зашифрованы только гомоморфным шифрованием. Данные возвращаются Сенсорному узлу.

В результате сенсорные узлы могут производить вычисления над данными, передаваемыми друг другу. Если требуется производить коррекцию зашифрованных данных (например, если сенсорные датчики выполняют измерение температуры, то можно таким же образом обработать и передать сенсорному узлу корректирующие дельты для измеряемых параметров). Основное преимущество разработанной модели состоит в том, что она уменьшает размер шифрованных данных, передаваемых между сенсорным узлом и базовой станцией. А также это сокращает количество ресурсов, требуемых на стороне сенсора.

2. Алгоритмы для представления базовых алгоритмов шифрования Кузнечик в виде КНФ

2.1 Алгоритм для построения полиномов Жегалкина по заданным S-блокам. Согласно Протоколу 1 данные, зашифрованные сначала симметричным шифром, а потом сверху гомоморфным шифром, можно расшифровать симметричной системой с использованием ключа шифрования, не снимая при этом гомоморфное шифрование. Для этого нужно представить все операции для алгоритма выработки раундовых ключей и для алгоритма расшифрования в виде простых булевых функций.

Для выполнения расшифрования симметричного шифра, закрытого гомоморфным шифром, необходимо представить функцию дешифрования в виде булевых функций или в виде арифметической схемы над заданным полем. Известно, что работу S-блока можно отобразить с использованием полиномов Жегалкина. В ходе работы был разработан и реализован алгоритм, который позволяет строить полиномы Жегалкина для заданного блока замены. Данный алгоритм представлен в работе [20]. Так как в алгоритме шифрования Кузнечик S-блок преобразует целый байт, то для его описания необходимо выработать 8 полиномов Жегалкина. Так, например, полином для получения бита x0 после прохождения через S-блок в алгоритме Кузнечик будут выглядеть так, как показано на рис. 6.

Полиномы для прямого и инверсного S-блоков замены в алгоритме шифрования S-KN1 представлены на рис. 7.

$$x0 = 1 + x1 + x0x1 + x2 + x1x2 + x0x1x2 + x3 + x1x3 + x0x2x3 + x4 + x0x4 + x0x1x4 + x2x4 + x0x2x4 + x1x2x4 + x1x3x4 + x0x1x2x3x4 + x5 + x0x5 + x0x1x5 + x2x5 + x0x2x5 + x0x1x2x5 + x3x5 + x1x3x5 + x0x1x3x5 + x0x2x3x5 + x1x2x3x5 + x0x1x2x3x5 + x4x5 + x1x4x5 + x0x1x2x4x5 + x2x3x4x5 + x0x2x3x4x5 + x0x1x2x3x4x5 + x1x6 + x2x6 + x0x1x2x6 + x0x3x6 + x0x2x3x6 + x0x1x2x3x6 + x4x6 + x0x4x6 + x2x4x6 + x0x1x2x4x6 + x3x4x6 + x0x1x3x4x6 + x0x2x3x4x6 + x0x1x5x6 + x2x5x6 + x0x1x2x5x6 + x1x3x5x6 + x0x1x3x5x6 + x0x1x2x3x5x6 + x0x4x5x6 + x1x4x5x6 + x2x4x5x6 + x0x2x4x5x6 + x0x1x3x4x5x6 + x0x1x2x3x4x5x6 + x0x7 + x1x7 + x0x1x7 + x2x7 + x0x1x2x7 + x1x3x7 + x0x1x3x7 + x1x2x3x7 + x0x1x2x3x7 + x4x7 + x2x4x7 + x1x2x4x7 + x0x1x2x4x7 + x3x4x7 + x0x3x4x7 + x2x3x4x7 + x0x2x3x4x7 + x0x1x2x3x4x7 + x5x7 + x0x1x5x7 + x2x5x7 + x1x2x5x7 + x0x1x2x5x7 + x0x1x3x5x7 + x1x2x3x5x7 + x0x1x2x3x5x7 + x1x4x5x7 + x0x2x4x5x7 + x1x2x4x5x7 + x0x1x2x4x5x7 + x0x3x4x5x7 + x0x1x3x4x5x7 + x1x2x3x4x5x7 + x0x6x7 + x1x6x7 + x0x1x6x7 + x2x6x7 + x0x1x2x6x7 + x3x6x7 + x0x3x6x7 + x1x3x6x7 + x4x6x7 + x0x4x6x7 + x2x4x6x7 + x0x2x3x4x6x7 + x1x2x3x4x6x7 + x0x1x2x3x4x6x7 + x5x6x7 + x0x5x6x7 + x0x2x5x6x7 + x1x2x5x6x7 + x0x1x2x5x6x7 + x0x3x5x6x7 + x0x1x3x5x6x7 + x0x1x2x3x5x6x7 + x4x5x6x7 + x1x4x5x6x7 + x0x1x4x5x6x7 + x2x4x5x6x7 + x0x2x4x5x6x7 + x3x4x5x6x7 + x0x3x4x5x6x7 + x2x3x4x5x6x7 + x0x2x3x4x5x6x7$$

Рис. 6. Один из полиномов Жегалкина для S-блока алгоритма Кузнечик

Прямой S-блок:

Вход $x_3x_2x_1x_0$	Выход $x'_3x'_2x'_1x'_0$
0000	0011
0001	0110
0010	1010
0011	0111
0100	1111
0101	0000
0110	0101
0111	1011
1000	0010
1001	1100
1010	0001
1011	1110
1100	0100
1101	1001
1110	1101
1111	1000

Для прямого S-блока:

$$x'_0 = 1 + x_0 + x_1 + x_3 + x_1x_2 + x_0x_3 + x_1x_2x_3 + x_0x_1x_2 + x_1x_2 + x_0x_3$$

$$x'_1 = 1 + x_0x_2 + x_1x_2 + x_0x_3 + x_1x_3 + x_2x_3$$

$$x'_2 = 0 + x_0 + x_2$$

$$x'_3 = 0 + x_1 + x_0x_1 + x_2 + x_0x_2 + x_0x_1x_2 + x_0x_3 + + x_1x_3 + x_0x_1x_3 + x_2x_3 + x_0x_2x_3 + x_1x_2x_3$$

Инверсный S-блок:

Вход $x_3x_2x_1x_0$	Выход $x'_3x'_2x'_1x'_0$
0000	0101
0001	1010
0010	1000
0011	0000
0100	1100
0101	0110
0110	0001
0111	0011
1000	1111
1001	1101
1010	0010
1011	0111
1100	1001
1101	1110
1110	1011
1111	0100

Для инверсного S-блока:

$$x'_0 = 1 + x_0 + x_1 + x_0x_1 + x_2 + x_0x_2 + x_0x_1x_2 + + x_0x_3 + x_2x_3 + x_1x_2x_3$$

$$x'_1 = 0 + x_0 + x_0x_1 + x_0x_1x_2 + x_3 + x_0x_2 + x_2x_3 + + x_1x_2x_3$$

$$x'_2 = 1 + x_0 + x_1 + x_0x_1 + x_0x_2 + x_0x_1x_2 + x_0x_3 + + x_2x_3 + x_1x_2x_3$$

$$x'_3 = 0 + x_1 + x_2 + x_0x_1x_2 + x_3 + x_0x_3 + x_2x_3 + x_1x_2x_3$$

Рис. 7. Один из полиномов Жегалкина для S-блока алгоритма S-KN1

Рассмотрим на примере алгоритма S-KN1 как можно реализовать полученные полиномы Жегалкина. Для этого необходимо определить какие из операций можно выполнять параллельно, и какая при этом будет использоваться глубина умножения (рис. 8).

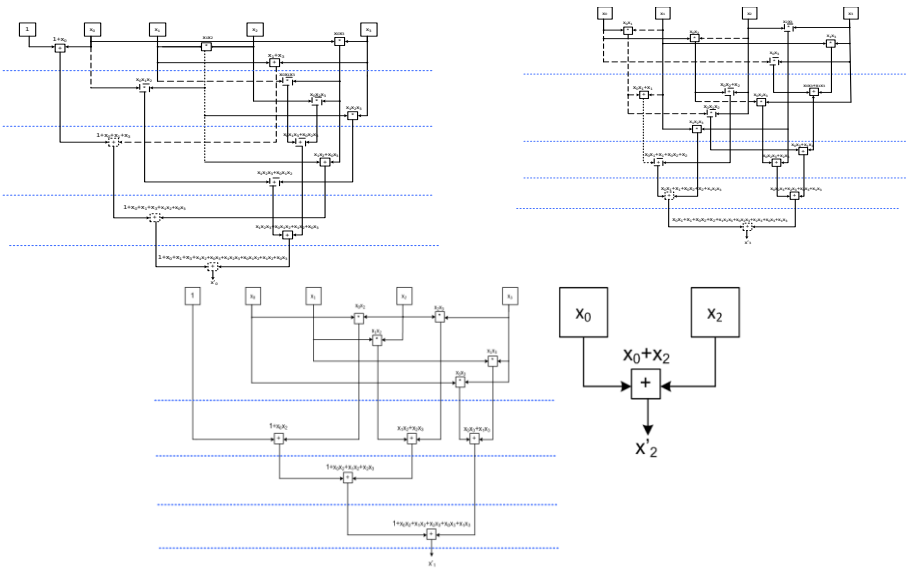

Рис. 8. Схемы реализации полиномов для алгоритма S-KN1

Так, из рис. 8 видно, что для бита x0 требуется 2 уровня умножения. Сложение по сравнению с умножением может вызвать переполнение только в одном разряде. Поэтому в данном случае глубиной сложения (=5) можно пренебречь. Для x1 требуется 1 уровень умножения, а для x2 требуется только сложение. Аналогичным образом строятся схемы и для других полиномов.

2.2 Алгоритм построения КНФ для линейного преобразования L. Линейное преобразование L заключается в полиномиальном перемножении констант со значениями данных в текущем блоке. Так как в случае гомоморфного шифрования каждый бит будет зашифрован по отдельности, то нам необходимо представить результат перемножения полиномов A и B по модулю F в виде сумм для битовых произведений (рис. 9). Можно видеть, что для получения каждого следующего значения слагаемого нам необходимо произвести сдвиг влево на 1 разряд в регистре множимого и добавить нормализующий полином, умноженный на старший разряд, вытолкнутый из регистра. В таком случае нормализация произойдет только в том случае, если вытолкнутый разряд был равен единицы. Каждый бит результата необходимо домножить на очередной бит множителя. Таким образом, для 4-разрябного полинома каждый бит результата будет получен в результате 7 умножений и 6 сложений (рис. 9).

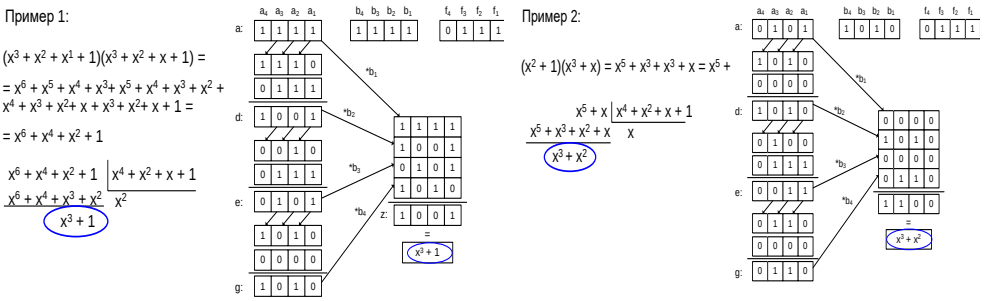

Рис. 9. КНФ для преобразования L в алгоритме S-KN1

Таким образом получаем 4 уровня умножения. Так как для применения операции L требуется применить полиномиальное перемножение дважды, то итоговый уровень умножения составит 8 шагов (рис. 10). Аналогичным образом формируются формулы для описания работы преобразования L оригинального шифра Кузнечик.

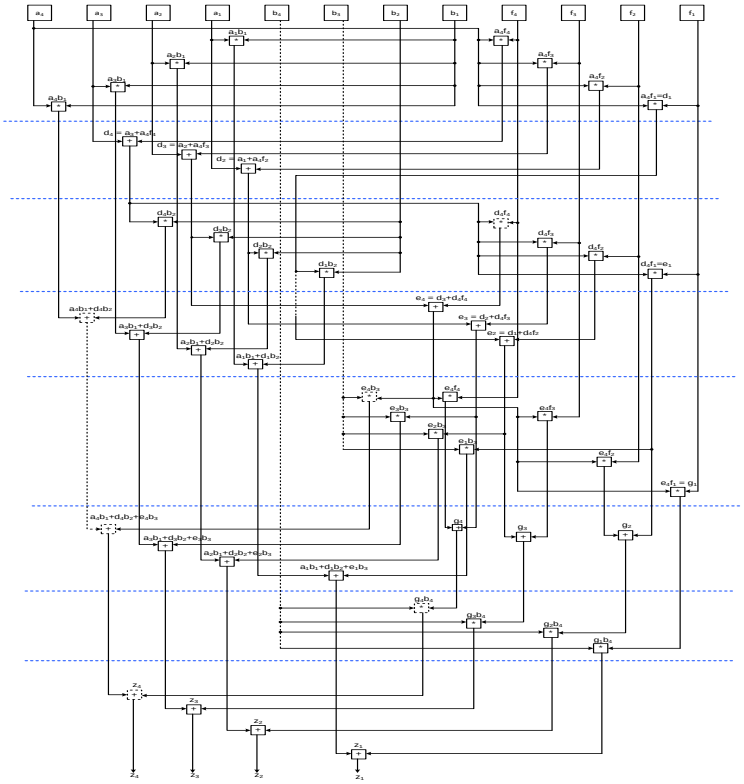

Рис. 10. Уровни умножения для побитного выполнения преобразования L

3. Алгоритмы работы разработанной гибридной схемы. На основании использования разработанных битовых умножений была разработана гибридная схема для расшифрования симметричного алгоритма с использованием гомоморфного шифрования. Схема состоит из двух частей: схемы выработки раундовых ключей и самой схемы расшифрования. На рис. 11 представлена схема выработки раундовых ключей. Каждый квадратик схемы – это один бит, зашифрованный гомоморфно. В результате применения разработанных схем мы получаем 4 раундовых ключа Key1, Key2, Key3, Key4.

Аналогичным образом была разработана схема для расшифрования данных S-KN1, зашифрованных гомоморфно. Каждый квадратик схемы – это один бит, зашифрованный гомоморфно. В результате применения разработанной схемы мы получаем 8 битов с z1 по z8, которые представляют собой открытые данные сенсорного узла, зашифрованные гомоморфно (рис. 12). Далее узел может обрабатывать эти данные (предпочтительно складывать в силу ограниченности ресурса сенсора) до тех пор, пока доверенное лицо не захочет эти данные считать и расшифровать. Аналогичным образом формируется алгоритм для оригинального шифра Кузнечик.

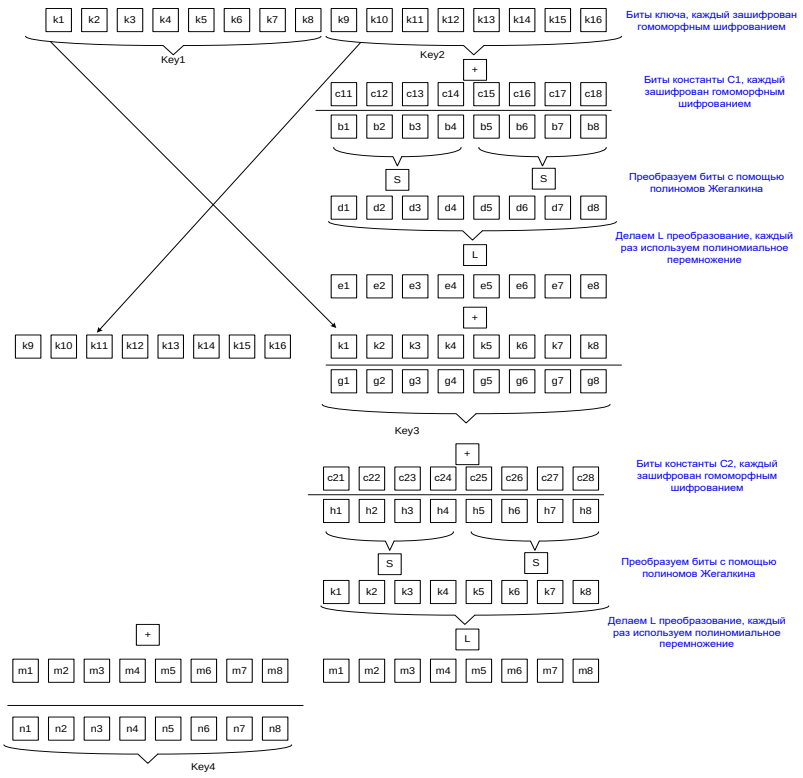

Рис. 11. Схема выработки раундовых ключей

Рис. 12. Схема расшифрования данных

Выводы. В данной статье описана разработка алгоритма гибридного шифрования на основе использования симметричного алгоритма шифрования Кузнечик и гомоморфной схемы BGV для возможности безопасной обработки и передачи информации в малоресурсных системах. Описаны алгоритмы выработки полиномов Жегалкина для S-блока замены алгоритма Кузнечик и S-KN1, схема представления преобразования L в виде последовательности простейших арифметических операций, гибридная схема для расшифрования симметричного алгоритма с использованием гомоморфного шифрования, схема выработки ключей. В результате применения предложенной гибридной возможна передача симметрично зашифрованных данных и гомоморфно зашифрованного ключа шифрования, при этом на принимающей стороне зашифрованное сообщение остается также зашифровано только гомоморфным шифром. При снятии симметричного шифрования информация будет недоступна обрабатывающему узлу. Все вышеперечисленное обеспечивает передачу гомоморфно зашифрованных данных между ресурсоемким вычислительным узлом и малоресурсными узлами для последующей обработки в зашифрованном виде.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Ноздрунов В.* Об уязвимостях протокола интернета вещей NB-Fi в новом проекте национального стандарта // Ежегодная международная научно-практическая конференция «РусКрипто’2021». – Режим доступа: https://www.ruscrypto.ru/resource/archive/rc2021/files/02_nozdrunov.pdf (дата обращения: 07.05.2021).
2. *Поликарпов А.* Особенности внедрения СКЗИ в РТК с БпЛА МД // Ежегодная международная научно-практическая конференция «РусКрипто’2021». – Режим доступа: https://www.ruscrypto.ru/resource/archive/rc2021/files/11_polikarpov.pdf (дата обращения: 07.05.2021).
3. *Полегенько А.* Способ сопряжения сетей разного уровня «открытости», организованных робототехническими комплексами и системами // Ежегодная международная научно-практическая конференция «РусКрипто’2021». – Режим доступа: https://www.ruscrypto.ru/resource/archive/rc2021/files/11_polegenko.pdf (дата обращения: 07.05.2021).
4. *Жуков А.* Легковесная криптография // Вопросы кибербезопасности. – 2015. – № 1 (9). – С. 26-44.
5. *Дерябин М.А., Кучеров Н.Н.* Обзор безопасных методов шифрования для облачных вычислений // Новости науки в АПК. – 2019. – № 3 (12). – С. 298-303.
6. *Mark A. Will, Ryan K.L.* The Cloud Security Ecosystem Chapter 5 - A guide to homomorphic encryption // Technical, Legal, Business and Management Issues. – 2015. – P. 101-127. – Режим доступа: https://www.sciencedirect.com/science/article/pii/B9780128015957000057 (дата обращения: 07.05.2021).
7. *Naehrig M., Lauter K.* Can Homomorphic Encryption be Practical? // Proceedings of the 3rd ACM Cloud Computing Security Workshop, CCSW 2011, Chicago, USA, 2011. – P. 113-124.
8. *Halevi S., Shoup V.* Design and implementation of HElib: a homomorphic encryption library. – Режим доступа: https://eprint.iacr.org/2020/1481 (дата обращения: 07.05.2021).
9. *Гаража А.А., Герасимов И.Ю., Николаев М.В., Чижов И.В.* Об использовании библиотек полностью гомоморфного шифрования // International Journal of Open Information Technologies. – 2021. – Vol. 9, No. 3. – P. 11-22.
10. *Gentry C.* Fully Homomorphic encryption using ideal lattices // Proceedings of 41-th ACM symposium on theory of computing (STOC). Bethesda, 2009. – P. 169-178.
11. *Gentry C., Halevi S., Smart N.P.* Better Bootstrapping in Fully Homomorphic Encryption Gentry // Public Key Cryptography – PKC 2012. Vol 7293 – 2012, Springer. – P. 1-16.
12. *Аракелов Г.Г., Грибов А.В., Михалев А.В.* Прикладная гомоморфная криптография: примеры // Фундаментальная и прикладная математика. – 2016. – Т. 21, № 3. – С. 25-38.
13. *Alam S., De D.* Analysis of Security Threats in Wireless Sensor Network // International Journal of Wireless & Mobile Networks. – 2014. – Vol. 6, No. 2, – P. 1-12.
14. *Borgohain T., Sanyal S.* Survey of Operating Systems for the IoT Environment // International Journal of Advanced Networking and Applications. – 2015. – Vol. 6. – P. 1-5.

15. *Gentry C., Halevi S., Smart N.P.* Homomorphic Evaluation of the AES Circuit // Advances in Cryptology - CRYPTO 2012. Vol 7417, Springer. – P. 850-867.
16. Криптографическая защита информации Блочные шифры – ГОСТ Р 34.12-2015. – URL: https://www.tc26.ru/standard/gost/GOST R 3412-2015.pdf (дата обращения: 07.05.2021).
17. *Бабенко Л.К., Ищукова Е.А., Толоманенко Е.А.* Дифференциальный анализ шифра Кузнечик // Известия ЮФУ. Технические науки. – 2017. – № 5 (190). – С. 25-37.
18. *Ishchukova E.A., Babenko L.K.* Two simplified versions of Kuznyechik cipher (GOST R 34.12-2015) // Proceedings of the 10h International Conference on Security of Information and Networks. – SIN '17. – New York, NY, USA: ACM, 2017.
19. *Brakerski Z., Gentry C.*, Vaikuntanathan Vinod. Fully homomorphic encryption without bootstrapping. // Cryptology ePrint Archive, Report 2011/277. – 2011. – Режим доступа: https://eprint.iacr.org/2011/277 (дата обращения: 07.05.2021)
20. *Babenko L.K., Tolomanenko E.A.* Development of algorithms for data transmission in sensor networks based on fully homomorphic encryption using symmetric Kuznyechik algorithm // Journal of Physics: Conference Series. – 2021. – Vol. 1812. – P. 246-251.

REFERENCES

1. *Nozdrunov V.* Ob uyazvimostyakh protokola interneta veshchey NB-Fi v novom proekte natsional'nogo standarta [About vulnerabilities of the Internet of Things protocol NB-Fi in the new draft of the national standard], *Ezhegodnaya mezhdunarodnaya nauchno-prakticheskaya konferentsiya «RusKripto'2021»* [Annual International scientific and practical Conference "RusCripto'2021"]. Available at: https://www.ruscrypto.ru/resource/archive/rc2021/files/02_nozdrunov.pdf (accessed 07 May 2021).
2. *Polikarpov A.* Osobennosti vnedreniya SKZI v RTK s BpLA MD [Features of the implementation of SKZI in RTC with UAV MD], *Ezhegodnaya mezhdunarodnaya nauchno-prakticheskaya konferentsiya «RusKripto'2021»* [Annual International scientific and practical Conference "RusCripto'2021"]. Available at: https://www.ruscrypto.ru/resource/archive/rc2021/files /11_polikarpov.pdf (accessed 07 May 2021).
3. *Polegen'ko A.* Sposob sopryazheniya setey raznogo urovnya «otkrytosti», organizovannykh robototekhnicheskimi kompleksami i sistemami [The method of interfacing networks of different levels of "openness", organized by robotic complexes and systems], *Ezhegodnaya mezhdunarodnaya nauchno-prakticheskaya konferentsiya «RusKripto'2021»* [Annual International scientific and practical conference "RusKripto ' 2021"]. Available at: https://www.ruscrypto.ru/ resource/archive/rc2021/files/11_polegenko.pdf (accessed 07 May 2021).
4. *Zhukov A.* Legkovesnaya kriptografiya [Lightweight Cryptography], *Voprosy kiberbezopasnosti* [Cybersecurity issues], 2015, No. 1 (9), pp. 26-44.
5. *Deryabin M.A., Kucherov N.N.* Obzor bezopasnykh metodov shifrovaniya dlya oblachnykh vychisleniy [Review of secure encryption methods for cloud computing], *Novosti nauki v APK* [Science news in the agro-industrial complex], 2019, No. 3 (12), pp. 298-303.
6. *Mark A. Will, Ryan K.L.* The Cloud Security Ecosystem Chapter 5 - A guide to homomorphic encryption, *Technical, Legal, Business and Management Issues*, 2015, pp. 101-127. Available at: https://www.sciencedirect.com/science/article/pii/B9780128015957000057 (accessed 07 May 2021).
7. *Naehrig M., Lauter K.* Can Homomorphic Encryption be Practical?, *Proceedings of the 3rd ACM Cloud Computing Security Workshop, CCSW 2011, Chicago, USA, 2011*, pp. 113-124.
8. *Halevi S., Shoup V.* Design and implementation of HElib: a homomorphic encryption library. Available at: https://eprint.iacr.org/2020/1481 (accessed 07 May 2021).
9. *Garazha A.A., Gerasimov I.Yu., Nikolaev M.V., Chizhov I.V.* Ob ispol'zovanii bibliotek polnost'yu gomomorfnogo shifrovaniya [On the use of fully homomorphic encryption libraries], *International Journal of Open Information Tech-nologies*, 2021, Vol. 9, No. 3, pp. 11-22.
10. *Gentry C.* Fully Homomorphic encryption using ideal lattices, *Proceedings of 41-th ACM symposium on theory of computing (STOC). Bethesda, 2009*, pp. 169-178.
11. *Gentry C., Halevi S., Smart N.P.* Better Bootstrapping in Fully Homomorphic Encryption Gentry, *Public Key Cryptography – PKC 2012. Vol 7293 – 2012, Springer*, pp. 1-16.
12. *Arakelov G.G., Gribov A.V., Mikhalev A.V.* Prikladnaya gomomorfnaya kriptografiya: primery [Applied homomorphic cryptography: examples], *Fundamental'naya i prikladnaya matematika* [Fundamental and applied mathematics], 2016, Vol. 21, No. 3, pp. 25-38.

13. *Alam S., De D.* Analysis of Security Threats in Wireless Sensor Network, *International Journal of Wireless & Mobile Networks*, 2014, Vol. 6, No. 2, pp. 1-12.
14. *Borgohain T., Sanyal S.* Survey of Operating Systems for the IoT Environment, *International Journal of Advanced Networking and Applications*, 2015, Vol. 6, pp. 1-5.
15. *Gentry C., Halevi S., Smart N.P.* Homomorphic Evaluation of the AES Circuit, *Advances in Cryptology - CRYPTO 2012. Vol 7417, Springer*, pp. 850-867.
16. Kriptograficheskaya zashchita informatsii Blochnye shifry – GOST R 34.12-2015 [Cryptographic protection of information Block ciphers-GOST R 34.12-2015]. Available at: https://www.tc26.ru/standard/gost/GOST R 3412-2015.pdf (accessed 07 May 2021).
17. *Babenko L.K., Ishchukova E.A., Tolomanenko E.A.* Differentsial'nyy analiz shifra Kuznechik [Differential analysis of the Grasshopper cipher], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2017, No. 5 (190), pp. 25-37.
18. *Ishchukova E.A., Babenko L.K.* Two simplified versions of Kuznyechik cipher (GOST R 34.12-2015), *Proceedings of the 10h International Conference on Security of Information and Networks. – SIN '17.* New York, NY, USA: ACM, 2017.
19. *Brakerski Z., Gentry C.,* Vaikuntanathan Vinod. Fully homomorphic encryption without bootstrapping, *Cryptology ePrint Archive, Report 2011/277*, 2011. Available at: https://eprint.iacr.org/2011/277 (accessed 07 May 2021)
20. *Babenko L.K., Tolomanenko E.A.* Development of algorithms for data transmission in sensor networks based on fully homomorphic encryption using symmetric Kuznyechik algorithm, *Journal of Physics: Conference Series*, 2021, Vol. 1812, pp. 246-251.

Статью рекомендовала к опубликованию к.т.н. Е.А. Ищукова.

Бабенко Людмила Климентьевна – Южный федеральный университет; e-mail: lkbabenko@sfedu.ru; г. Таганрог, Россия; тел.: 88634312018; кафедра безопасности информационных технологий; профессор.

Толоманенко Екатерина Алексеевна – e-mail: kat.tea@mail.ru; тел.: 88634371905; кафедра безопасности информационных технологий; аспирант.

Babenko Lyudmila Klimentevna – Southern Federal University; e-mail: lkbabenko@sfedu.ru; Taganrog, Russia; phone: +78634312018; the department of security of information technologies; professor.

Tolomanenko Ekaterina Alekseevna – e-mail: kat.tea@mail.ru; phone: +78634371905; the department of security of information technologies; postgraduate student.

УДК 612.743, 612.817.2 DOI 10.18522/2311-3103-2021-2-18-31

Н.А. Будко, М.Ю. Медведев, А.Ю. Будко

РАЗРАБОТКА И ИССЛЕДОВАНИЕ МЕТОДА ВЕКТОРНОГО АНАЛИЗА ЭМГ ПРЕДПЛЕЧЬЯ ДЛЯ ПОСТРОЕНИЯ ЧЕЛОВЕКО-МАШИННЫХ ИНТЕРФЕЙСОВ

Рассматриваются проблемы увеличения глубины и повышения долговременной устойчивости каналов связи в интерфейсах человек-машина, построенных на основе данных об электрической активности мышц предплечья. Возможным вариантом решения является применение метода анализа сигналов электромиограмм (ЭМГ), совмещающий векторное и командное управление. В виду возможности случайного смещения положения электродов в процессе эксплуатации, построена математическая модель для векторного анализа ЭМГ в сферических координатах, инвариантная к пространственному расположению электродов на предплечье. Командное управление осуществляется на основе распознавания жестов посредством предварительно обученной искусственной нейронной сети (ИНС). Векторное управление заключается в решении задачи калибровки каналов датчиков ЭМГ по пространственному расположению электродов и расчета результирующего вектора мы-