15. *Nazarov A.A., Terpugov A.F.* Teoriya massovogo obsluzhivaniya [Queuing theory]. Tomsk: Izd-vo NTL, 2010, 228 p.
16. *Saati T.L.* Elementy teorii massovogo obsluzhivaniya i ee prilozheniya [Elements of the theory of queuing and its applications]. 3 ed. Moscow: Knizhnyy dom «LIBROKOM», 2010, 520 p.
17. *Kleynrok L.* Teoriya massovogo obsluzhivaniya [Theory of queuing]. Moscow: Mashinostroenie, 1979, 432 p.
18. *Borovkov A.A.* Veroyatnostnye protsessy v teorii massovogo obsluzhivaniya [Probabilistic processes in the theory of queuing]. Moscow: Nauka, 1972, 368 p.
19. *Venttsel' E.S.* Teoriya sluchaynykh protsessov i ee inzhenernye prilozheniya [Theory of random processes and its engineering applications]. Moscow: Nauka, 1991, 384 p.
20. *Feller V.* Vvedenie v teoriyu veroyatnostey i ee prilozheniya [Introduction to probability theory and its applications]: In 2 vol. Vol. 1. Moscow: LIBROKOM, 2010, 528 p.

Статью рекомендовал к опубликованию д.т.н., профессор Б.М. Глинский.

Павский Валерий Алексеевич – Кемеровский государственный университет (КемГУ); e-mail: pavva46@mail.ru; г. Кемерово, Россия; тел.: +73842396832; д.т.н.; профессор; профессор кафедры общей математики и информатики

Павский Кирилл Валерьевич – Институт физики полупроводников им. А.В. Ржанова СО РАН (ИФП СО РАН); e-mail: pkv@isp.nsc.ru; г. Новосибирск, Россия; тел.: +73833332171, 3305626; д.т.н.; доцент; зав. лабораторией вычислительных систем; профессор кафедры вычислительных систем Сибирского государственного университета телекоммуникаций и информатики (СибГУТИ).

Pavsky Valery Alekseevich – Kemerovo State University; e-mail: pavva46@mail.ru; Kemerovo, Russia; phone: +73842396832; dr. of eng. sc.; professor; professor of department of general mathematics and informatics.

Pavsky Kirill Valerievich – Rzhanov Institute of Semiconductor Physics Siberian Branch of Russian Academy of Sciences; e-mail: pkv@isp.nsc.ru; Novosibirsk, Russia; phone: +73833332171, 3305626; dr. of eng. sc.; head of computer systems laboratory; professor of computer systems department, SibSUTIS.

УДК 004.056 DOI 10.18522/2311-3103-2022-4-103-112

И.Д. Русаловский, Л.К. Бабенко, О.Б. Макаревич

РАЗРАБОТКА МЕТОДОВ ГОМОМОРФНОГО ДЕЛЕНИЯ*

Рассматриваются проблемы гомоморфной криптографии. Гомоморфная криптография – одно из молодых направлений криптографии. Его отличительная особенность заключается в том, что можно обрабатывать зашифрованные данные без их предварительной расшифровки таким образом, что результат операций над зашифрованными данными эквивалентен после расшифровки результату операции над открытыми данными. Гомоморфное шифрование может эффективно применяться для реализации защищенных облачных вычислений. Для решения различных прикладных задач требуется поддержка всех математических операций, в том числе и операции деления, однако эта тема недостаточно проработана. Возможность выполнить операцию деления гомоморфно позволит расширить возможности прикладного применения гомоморфного шифрования и позволит выполнить гомоморфную реализацию многих алгоритмов. В работе рассматриваются существующие гомоморфные алгоритмы и возможность реализации операции деления в рамках этих алгоритмов. Также в работе предлагаются два метода гомоморфного деления. Первый метод основан на представлении шифротекстов в виде простых дробей и

* Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 20-37-90140.

выражении операции деления через операцию умножения. В рамках второго метода предлагается представление шифротекстов в виде массива гомоморфно зашифрованных бит, а все операции, в том числе и рассматриваемую в данной статье операцию деления, выражать через бинарные гомоморфные операции. Рассматриваются возможные подходы к реализации деления через бинарные операции и выбирается подход, наиболее подходящий для гомоморфной реализации. Выполняется анализ предложенных методов и указываются их преимущества и недостатки.

Гомоморфное шифрование; криптографическая защита; методы и алгоритмы; гомоморфное деление.

I.D. Rusalovsky, L.K. Babenko, O.B. Makarevich

DEVELOPMENT OF HOMOMORPHIC DIVISION METHODS

The article deals with the problems of homomorphic cryptography. Homomorphic cryptography is one of the young areas of cryptography. Its distinguishing feature is that it is possible to process encrypted data without decrypting it first, so that the result of operations on encrypted data is equivalent to the result of operations on open data after decryption. Homomorphic encryption can be effectively used to implement secure cloud computing. To solve various applied problems, support for all mathematical operations, including the division operation, is required, but this topic has not been sufficiently developed. The ability to perform the division operation homomorphically will expand the application possibilities of homomorphic encryption and will allow performing a homomorphic implementation of many algorithms. The paper considers the existing homomorphic algorithms and the possibility of implementing the division operation within the framework of these algorithms. The paper also proposes two methods of homomorphic division. The first method is based on the representation of ciphertexts as simple fractions and the expression of the division operation through the multiplication operation. As part of the second method, it is proposed to represent ciphertexts as an array of homomorphically encrypted bits, and all operations, including the division operation considered in this article, are implemented through binary homomorphic operations. Possible approaches to the implementation of division through binary operations are considered and an approach is chosen that is most suitable for a homomorphic implementation. The proposed methods are analyzed and their advantages and disadvantages are indicated.

Homomorphic encryption; cryptographic protection; methods and algorithms; homomorphic division.

Введение. В современном мире информационные технологии активно применяются во всех сферах жизни. В результате процесса информатизации вырос объем информации, возросли информационные потоки. В условиях бурного роста информационных технологий как никогда ранее стала актуальна проблема обеспечения информационной безопасности, в частности проблема обеспечения конфиденциальности информации. Актуальность необходимости обеспечения конфиденциальности информации обострилась с появлением и широким распространением облачных технологий. Классические криптографические средства обеспечивают необходимый уровень защиты данных при их передаче от клиента на облачный сервис по незащищенному каналу связи. Однако после передачи сервис получает неограниченный доступ к данным клиента для обработки. В этом кроется потенциальная уязвимость, так как сервис может быть оказаться недобросовестным, либо может быть подменен или скомпрометирован. Для решения этой проблемы может быть применена гомоморфная криптография [1–7]. Гомоморфное шифрование – это особый вид шифрования, позволяющий выполнять операции над зашифрованными данными и получать зашифрованный результат, соответствующий результату выполнения операции над открытыми данными. Эта особенность позволяет эффективно применять данный вид шифрования для решения любых задач

обработки данных без раскрытия самих данных, что актуально, например, для реализации защищенных облачных вычислений и защищенного поиска информации. Данные в этом случае шифруются на стороне клиента, а передаются и обрабатываются в зашифрованном виде. Ключ расшифрования при этом никуда не передается, следовательно, при соблюдении криптографической стойкости алгоритма выполнить расшифрование может только клиент.

Анализ актуальности. Для решения прикладных задач необходима поддержка как можно большего числа гомоморфных операций над целыми числами – сложение, разность, умножение и деление. Одной из часто решаемых прикладных задач является решение систем линейных алгебраических уравнений (СЛАУ). Решение СЛАУ требуется во многих задачах и алгоритмах, например, используется для нахождение некоторых неизвестных коэффициентов на основе ряда экспериментов. При решении подобной задачи для каждого эксперимента строится линейное алгебраическое уравнение, а неизвестные коэффициенты вычисляются на основе решения СЛАУ порядка n, где n – число поставленных экспериментов. Одним из наиболее популярных методов решения СЛАУ является алгоритм Гаусса. Для выполнения его гомоморфной реализации необходима поддержка следующих гомоморфных операций: сложение, разность, умножение, деление, сравнение (для исключения нулевых элементов с главной диагонали). Также операция гомоморфного деления необходима для гомоморфной реализации других алгоритмов.

На данный момент существует большое количество алгоритмов полностью гомоморфного шифрования, основанных на различных принципах [8–16]. Для ряда из них выполнены практические реализации, которые находятся в общем доступе, есть и коммерческие продукты. Однако ни в одной из найденных во время анализа реализаций не было поддержки операции деления. Таким образом, актуальным является разработка метода или алгоритма, позволяющего расширить существующие гомоморфные алгоритмы функциональностью деления.

Операция деления является обратной к операции умножения. Следовательно, чтобы реализовать гомоморфное деление гомоморфный алгоритм шифрования должен проявлять мультипликативные свойства. Рассмотрим несколько алгоритмов гомоморфного шифрования и проанализируем возможность выполнения операции деления на их базе, а также предложим методы реализации гомоморфного деления.

Алгоритмы в кольце вычетов. Одним из вариантов построения гомоморфных алгоритмов является отображение в кольце вычетов. Примером такого алгоритма является RSA. Алгоритм RSA проявляет мультипликативный гомоморфизм, а в кольце вычетов можно найти обратный элемент. Следовательно, возможно реализовать операцию деления как умножение на обратное. Однако, операция деления во множестве действительных чисел R и в кольце вычетов Z_n не во всех случаях эквивалентна. Результатом выполнения операции деления во множестве целых чисел будут частное и остаток, и частное будет округлено до целого числа и точность результата деления будет снижена. Однако для решения некоторых задач было бы достаточно деления с низкой точностью, в результате которого остаток бы полностью отбрасывался.

Рассмотрим кольцо Z_5 в качестве примера. Обратный элемент можно найти, воспользовавшись следствием из формулы Эйлера (1):

$$m^{-1} \bmod p \equiv m^{p-2} \bmod p. \quad (1)$$

Продемонстрируем несколько численных примеров. Пусть $m_1 = 4$, $m_2 = 2$, тогда: $\frac{m_1}{m_2} \bmod 5 \equiv \frac{4}{2} \bmod 5 \equiv 4 * 2^{-1} \bmod 5 \equiv 4 * 3 \bmod 5 \equiv 2 \bmod 5$

В результате получаем, что 4 / 2 = 2, что верно. Однако, в рассматриваемой схеме операции выполняются в кольце, в то время как для прикладного использования нас интересуют целочисленные операции во множестве целых чисел. В качестве наглядного примера разделим 2 на 4:

$$\frac{m_1}{m_2} \; mod \; 5 \; \equiv \frac{2}{4} \; mod \; 5 \; \equiv 2 * \; 4^{-1} \; mod \; 5 \; \equiv 2 * 4 \; mod \; 5 \; \equiv 3 \; mod \; 5.$$

Как видно из примера, 2 / 4 = 3 в кольце Z_5, в то время как мы ожидали получить 0 или 1, в зависимости от выбранной стратегии округления результата. Следовательно, данное решение не подходит для реализации гомоморфного деления.

Алгоритмы на основе полиномов. Еще один вариант реализации гомоморфного алгоритма шифрования – соотнесение открытому тексту некоторого полинома. К примеру, Ф. Буртыка предложил алгоритм шифрования на основе матричных полиномов (полиномов, каждый коэффициент которых представлен матрицей) [17]. Также в выпускной квалификационной работе Яковлева [18] предлагается алгоритм шифрования посредством преобразования целого числа полиному с целочисленными коэффициентами. Между двумя полиномами можно выполнить операцию деления, результатом которой будут частное и остаток от деления. Как было указано в предыдущем примере, для решения некоторых задач может хватить точности деления, при которой остаток полностью отбрасывается. Однако в случае с шифротекстами на основе полиномов возникает ряд проблем.

Величина открытого текста никак не связана с порядком полинома, следовательно большему числу может соответствовать меньший полином и наоборот, а делимое полностью будет остатком.

Результатом деления будут частное и остаток, каждый из которых представлен полиномом. Если расшифровать их, разделить расшифрованный остаток на расшифрованный делитель, то мы получим корректный результат. Но частное и остаток в зашифрованном виде не соответствуют частному и остатку в расшифрованном виде. Таким образом, остаток от деления в зашифрованном виде может содержать большую часть частного, что делает операцию отбрасывания остатка некорректной, но не отбросить остаток мы не можем, так как в рамках алгоритма могут обрабатываться только полиномы.

Рассмотрим численный пример на основе алгоритма Яковлева. Пусть даны целые числа $m_1 = 4$, $m_2 = 1$, $p = 4$, $q = 2$, $x_0 = p / q = 2$ – секретный ключ. Выполним шифрование:

$f_1(x) = 5x + 2$; $f_1(x_0) = 12$

$g_1(x) = 22 * f_1(x) – 22 * 12 + m_1 = 20x + 8 – 48 + 4 = 20x - 36$

$f_2(x) = 3x – 5$; $f_2(x_0) = 1$

$g_2(x) = 22 * f_2(x) – 22 * 1 + m_2 = 12x – 20 – 4 + 1 = 12x - 23$

В результате деления получим $20x – 36$ на $12x – 23$ получим $g_3(x)) = 1$, остаток $g_4(x) = 8x-13$. После расшифрования получим:

$D(g_3(x)) = g_3(x_0) = 1$

$D(g_4(x)) = g_4(x_0) = 8 * 2 – 13 = 3$

Таким образом в результате деления получаем 1+3=4, однако на остаток пришлось 3, из-за чего мы не можем отбросить остаток от деления, а следовательно, продолжать вычисления без перезашифрования результата.

Метод гомоморфного деления на основе представления шифротекстов в виде простых дробей. Данный метод [19] позволяет реализовать гомоморфное деление на основе любого полностью гомоморфного алгоритма шифрования над целыми числами. Метод хорошо подходит для программной реализации и на его

основе может быть построен программный комплект, позволяющий реализовать математический аппарат для вычислений в рамках гомоморфной криптографии. С помощью предлагаемого метода можно шифровать не только целые, но и рациональные числа Q. В случае, если шифруется рациональное число, первый пункт можно пропустить.

Пусть дан некоторый полностью гомоморфный алгоритм шифрования над целыми, для которого определены E(m) – алгоритм шифрования, D(c) – алгоритм расшифрования, обратный к E(m), $\otimes$,$\oplus$ – операторы гомоморфного умножения и сложения над зашифрованными данными соответственно, где m – открытый текст, c – шифротекст. Тогда схема шифрования целого числа m с поддержкой операции деления может быть построена следующим образом:

1. Представляем шифруемый открытый текст в виде рационального числа: $m = \frac{m_1}{m_2}$, если число m – целое, то $m_2 = 1$.
2. Шифруем делимое и делитель полученного рационального числа: $a_1 = E(m_1)$, $b_2 = E(m_2)$.
3. Шифротекст в предлагаемой схеме шифрования будет представлен в виде пары шифротекстов: $C = (a_1; b_2)$.

Алгоритм расшифрования:

1. Расшифруем гомоморфно зашифрованные делимое и делитель, в виде которых представлен шифротекст: $m_1 = D(a_1)$; $m_2 = D(b_2)$
2. Результат расшифрования будет равен рациональному числу $m = \frac{m_1}{m_2}$. Полученный результат можно при необходимости округлить до целого.

Реализация математических операций:

1. Сложение. $C_1 + C_2 = (a_1 \otimes b_2 \oplus a_2 \otimes b_1;\ b_1 \otimes b_2)$.
2. Умножение. $C_1 * C_2 = (a_1 \otimes a_2;\ b_1 \otimes b_2)$.
3. Деление. $C_1/C_2 = (a_1 \otimes b_2;\ b_1 \otimes a_2)$.

К преимуществам данного подхода можно отнести возможность реализации операции гомоморфного деления на основе любого полностью гомоморфного алгоритма шифрования над целыми числами. Сама операция деления выполняется только над конечным результатом единожды на шифрующей стороне (на стороне клиента). Поэтому при гомоморфных вычислениях отсутствует округление промежуточных результатов, поэтому данный метод обеспечивает высокую точность вычислений. Конечно, предлагаемый подход имеет негативные стороны – увеличение объема шифротекста и сложности вычислений. Размер шифротекста увеличивается вдвое, так как вместо одного гомоморфно зашифрованного числа мы храним пару. Сложность вычислений тоже возрастает – сложность операции умножения возрастает в два раза, сложность операции деления эквивалентна сложности операции умножения, а сложность операции сложения увеличивается приблизительно в четыре раза. Однако возможность выполнения операции деления сильно расширяет возможности прикладного применения гомоморфного шифрования, что нивелирует снижение скорости вычислений.

Метод гомоморфного деления на основе битовых операций. Гомоморфное деление возможно реализовать через битовые операции на основе некоторого полностью гомоморфного алгоритма шифрования над битами. Для этого рассмотрим возможные подходы к выполнению операции деления над числами, представленными в двоичном коде. Как правило операция деления строится на последовательных операциях разности и сдвигов и возможны два варианта:

- деление со сдвигом делителя вправо;
- деление со сдвигом делимого влево.

Рассмотрим каждый из подходов и проанализируем, какой из них лучше подходит для гомоморфной реализации.

Алгоритм деления со сдвигом делителя вправо:

1. Проверка делителя на ноль. Это необходимо для корректной работы алгоритма, иначе выполнить шаг 2 будет невозможно. В случае, если делитель равен нулю, алгоритм завершается ошибкой.
2. Нормализация делителя. Для выполнения деления необходимо, чтобы старший разряд делителя был значащим, то есть содержал единицу. Для достижения этого мы предварительно сдвигаем делитель влево необходимое число раз, а количество сдвигов k запоминаем. Шаги 3-5 мы будем повторять k+1 число раз.
3. Вычитаем делитель из делимого и получаем текущий остаток. На основе комбинации знаков делимого, делителя и текущего остатка формируем очередной разряд частного. Также на основе комбинации знаков либо восстанавливается предыдущий остаток, либо используется текущий.
4. Сдвиг делителя на один разряд вправо
5. Уменьшаем значение счетчика итераций на 1 и проверяем его значение. Если он стал равен нулю, то завершаем процесс деления, иначе возвращаемся к шагу 3.

Плюсом данного подхода является оптимизация процесса деления за счет нормализации делителя и уменьшения числа итераций. Но это является огромным недостатком при построении гомоморфной реализации данного алгоритма, так как итерации выполняются внутри внешней системы, а информация о количестве сдвигов зашифрована гомоморфно. Из-за этого мы не можем получить информацию о числе шагов и придется выполнять максимально возможное число шагов гомоморфно, что значительно усложнит реализацию алгоритма. Поэтому рассмотрим вариант деления со сдвигом делимого.

Алгоритм деления со сдвигом делимого влево:

1. Формируем начальный остаток
2. Сдвигаем содержимое делимого влево, вдвигая выдвигаемый старший разряд в остаток.
3. Вычитаем делитель из остатка. Фактически, это вычитание из старших разрядов делимого, которые мы постепенно перемещаем на место остатка. Как и в первом варианте алгоритма, в зависимости от знака результата формируются биты частного и при необходимости выполняется восстановление остатка.
4. Повторяем шаги 2-3, пока не будут сформированы n+1 битов частного (где n – разрядность частного)
5. Прибавляем к частному единицу для округления результата.

Данный подход лучше подходит для гомоморфной реализации, так как не требует нормализации и выполняется всегда за фиксированное число шагов. Также алгоритм может обработать ноль в качестве делителя, однако в этом случае в частном будет некоторое ошибочное значение. Поэтому проверка на ноль и генерация ошибки необходима в любом случае. На основе данного подхода предложим метод гомоморфного деления на основе гомоморфных бинарных операций.

Пусть дан некоторый полностью гомоморфный алгоритм шифрования над битами, для которого определены E(m) – алгоритм шифрования, D(c) – алгоритм расшифрования, обратный к E(m), $\otimes$,$\oplus$ – гомоморфные логические операции, определённые в данном алгоритме шифрования, где m – открытый текст, c – шифротекст.

В общем виде алгоритм шифрования можно представить следующим образом:

1. Целое число m представляем побитно: $m = m_1 m_2 \ldots m_n$.
2. Каждый бит шифруется отдельно: $C = E(m_1)E(m_2)\ldots E(m_n)$.

Алгоритм битового деления будет аналогичен алгоритму машинного деления, с той лишь разницей, что обрабатываются гомоморфно зашифрованные биты, из-за чего будут некоторые особенности реализации. Также может возникнуть выразить недостающие логические операции через те, которые поддерживаются гомоморфной схемой шифрования. Как было рассмотрено с статье [20], операцию гомоморфного сравнения возможно реализовать на основе любого полностью гомоморфного алгоритма шифрования над битами, поэтому реализация алгоритма деления возможна. Операции суммы, разности и умножения в предлагаемой криптосистеме будут рассмотрены в другой статье.

Для облегчения выполнения операции разности в данной криптосистеме числа представляются в дополнительном коде. Алгоритм гомоморфного деления целых чисел, представленных в виде массива гомоморфно зашифрованных битов, можно представить следующим образом:

1. Нормализация делимого и делителя. В случае, если делитель и делимое имеют разную разрядность, нормализуем их разрядность к одной величине – наибольшей разрядности среди них. В шифротекст с меньшей разрядностью для нормализации необходимо продублировать самый левый разряд (знаковый) необходимое число раз.
2. Определяем разрядность частного. Минимально допустимое значение модуля делителя равно 1, поэтому числитель по модулю не может быть больше делимого. Следовательно, разрядность частного будет равна разрядности делимого.
3. Формируем начальный остаток, заполняя его знаковым битом делителя.
4. Сдвигаем содержимое делимого влево, игнорируя знаковый бит. Выдвинутый бит вдвигаем справа в промежуточный остаток.
5. Прибавляем модуль делителя к остатку, взяв его с противоположным знаком. Фактически, это вычитание из старших разрядов делимого, которые мы постепенно перемещаем на место остатка. Если знак полученного остатка совпадает со знаком делителя, то значение следующего бита частного устанавливаем равным 1, иначе 0. Если знак полученного остатка совпадает со знаком делимого, то используем этот остаток на следующей итерации, иначе выполняем операцию «восстановления» остатка – используем предыдущее значение остатка.
6. Если сформированы n+1 битов частного, где n – разрядность частного, то переходим к следующему шагу. Иначе возвращаемся к шагу 4.
7. Прибавляем к частному единицу округления. В случае, если мы получаем отрицательный результат, начальное заполнение частного будет в обратном коде. Поэтому прибавление единицы автоматически округляет результат и переводит его в дополнительный код, в случае отрицательного результата.
8. Отбрасываем (n+1)-й бит частного. Полученное частное является результатом деления.

Криптосистема на основе битовых операций обладает рядом преимуществ. Самое главное заключается в том, что в рамках данной криптосистемы можно реализовать практически любой алгоритм гомоморфно, так как поддерживаются все арифметические и логические операции. Однако сложность операций высока ввиду использования гомоморфного шифрования над битами, и размеры шифротекстов сильно увеличиваются. Также важно учитывать, что гомоморфное деление в рамках данной криптосистемы будет выполняться с округлением.

Заключение. В рамках данной статьи рассмотрена проблема гомоморфного деления целых чисел, предложены и описаны методы реализации гомоморфного деления. Практическая ценность работы состоит в решении одной из проблем гомоморфного шифрования – реализация гомоморфного деления, что позволяет расширить область практического применения гомоморфного шифрования. Метод

гомоморфного деления может использоваться в ряде алгоритмов, например, для нахождения среднего арифметического нескольких чисел или решения СЛАУ методом Гаусса.

Предложенные методы обладают как плюсами, так и минусами, поэтому выбор метода должен опираться на общую постановку задачи, которая должна быть решена с применением гомоморфной криптографии. В дальнейшем планируется рассмотреть возможности повышения эффективности предложенных методов, а также рассмотреть возможность обработки чисел с фиксированной или плавающей запятой.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Батура Т.В., Мурзин Ф.А., Семич Д.Ф.* Облачные технологии: основные понятия, задачи и тенденции развития // Программные продукты и системы. – 2014. – № 3. – С. 64-72.
2. *Жиров А.О., Жирова О.В., Кренделев С.Ф.* Безопасные облачные вычисления с помощью гомоморфной криптографии // Безопасность информационных технологий. – 2013. – Т. 1. – С. 6-12.
3. *Беккер М.Я., Гатчин Ю.А., Кармановский Н.С., Терентьев А.О., Федоров Д.Ю*. Информационная безопасность при облачных вычислениях: проблемы и перспективы // Научно-технический вестник информационных технологий, механики и оптики. – 2011. – С. 97-102.
4. *Денисов Д.В.* Перспективы развития облачных вычислений // Прикладная информатика. – 2009. – № 5 (23). – С. 52-58.
5. *Ковалев Д.* Информационная безопасность облачных вычислений // T-Comm. – 2011. – № S1. – С. 14-16.
6. *Трубей А.И.* Гомоморфное шифрование: безопасность облачных вычислений и другие приложения (обзор) // Информатика. – 2015. – № 1 (45). – С. 90-101.
7. *Бабенко Л.К., Буртыка Ф.Б., Макаревич О.Б., Трепачева А.В.* Защищенные вычисления и гомоморфное шифрование. // III Национальный суперкомпьютерный форум (25-27 ноября 2014, г. Переславль-Залесский). – ИПС им. А.К. Айламазяна РАН, 2014.
8. *Gentry C.* A Fully homomorphic encryption using ideal lattices // Symposium on the Theory of Computing (STOC). – Bethesda, USA, 2009. – P. 169-178.
9. *Gentry C., Sahai A., Waters B.* Homomorphic Encryption from Learning With Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based // Advances in cryptology – CRYPTO-2013, 33rd Annual Cryptology Conf. – Santa Barbara, CA, USA, 2013. – Part 1. – P. 73-93.
10. *Parmar P.V.* Survey of various homomorphic encryption algorithms and schemes // Intern. J. of Computer Applications. – 2014. – Vol. 91, No. 8. – P. 26-32.
11. *Jain N., Pal S.K., Upadhyay D.K.* Implementation and analysis of homomorphic encryption schemes // Intern. J. on Cryptography and Information Security (IJCIS). – 2012. – Vol. 2, No. 2. – P. 27-44.
12. *Smart N., Vercauteren F.* Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes // Public Key Cryptography – PKC 2010: 13th International Conference on Practice and Theory in Public Key Cryptography, Paris, France, May 26-28, 2010, Proceedings / P.Q. Nguyen, D. Pointcheval. – Berlin, Heidelberg, New York, NY, London [etc.]: Springer Science+Business Media, 2010. – P. 420-443.
13. *Smart N., Vercauteren F.* Fully homomorphic SIMD operations // Des. Codes Cryptogr. – Springer US, Springer Science+Business Media, 2014. – Vol. 71, Iss. 1. – P. 57-81.
14. *Gentry C., Halevi S.* Implementing Gentry's Fully-Homomorphic Encryption Scheme // Advances in Cryptology – EUROCRYPT 2011: 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011, Proceedings / K.G. Paterson. – Springer Science+Business Media, 2011. – P. 129-148.
15. *Dijk M. v., Gentry C., Halevi S., Vaikuntanathan V.* Fully Homomorphic Encryption over the Integers // Advances in Cryptology – EUROCRYPT 2010: 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, French Riviera, May 30 - June 3, 2010. Proceedings / H. Gilbert. – Berlin: Springer Berlin Heidelberg, 2010. – P. 24-43.
16. *Coron J., Mandal A., Naccache D., Tibouchi M.* Fully Homomorphic Encryption over the Integers with Shorter Public Keys // Advances in Cryptology – CRYPTO 2011: 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011, Proceedings / P. Rogaway. – Springer Science+Business Media, 2011. – P. 487-504.

17. *Буртыка Ф.Б.* Пакетное симметричное полностью гомоморфное шифрование на основе матричных полиномов // Тр. Института системного программирования РАН. – 2014. – Т. 26. № 5. – С. 99-116.
18. *Яковлев М.О.* Защищенный калькулятор. Разработка клиентского компонента // Выпускная квалификационная работа бакалавра. – URL: http://www.nsu.ru/xmlui/bitstream/handle/nsu/471/Text_YakovlevMO.pdf (дата обращения 20.09.2022).
19. *Бабенко Л.К., Русаловский И.Д.* Метод реализации гомоморфного деления // Известия ЮФУ. Технические науки. – 2020. – № 4 (214). – С. 212-221.
20. *Бабенко Л.К., Русаловский И.Д.* Масштабирование цифровых изображений с применением гомоморфного шифрования // Вопросы кибербезопасности. – 2021. – № 3(43). – С. 2-10.

REFERENCES

1. *Batura T., F.A. Murzin, D.F.* Semich Oblachnye tekhnologii: osnovnye ponyatiya, zadachi i tendentsii razvitiya [Cloud technologies: basic concepts, tasks and development trends], *Programmnye produkty i sistemy* [Software products and systems], 2014, No. 3, pp. 64-72.
2. *Zhirov A.O., Zhirova O.V., Krendelev S.F.* Bezopasnye oblachnye vychisleniya s pomoshch'yu gomomorfnoy kriptografii [Secure Cloud Computing with Homomorphic Cryptography], *Bezopasnost' informacionnykh tekhnologiy* [Information technology security], 2013, No. 1, pp. 6-12.
3. *Bekker M., Gatchin Yu., Karmanovskiy N., Terent'ev A., Fedorov D.* Informatsionnaya bezopasnost' pri oblachnykh vychisleniyakh: problemy i perspektivy [Information security in cloud computing: problems and prospects], *Nauchno-tekhnicheskiy vestnik informatsionnykh tekhnologiи, mekhaniki i optiki* [Scientific and technical bulletin of information technologies, mechanics and optics], 2011, pp. 97-102.
4. *Denisov D.* Perspektivy razvitiya oblachnykh vychisleniy [Prospects for the development of cloud computing], *Prikladnaya informatika* [Applied Informatics], 2009, No. 5 (23), pp. 52-58.
5. *Kovalev D.* Informatsionnaya bezopasnost' oblachnykh vychisleniy [Information security of cloud computing], *T-Comm*, 2011, No. S1, pp. 14-16.
6. *Trubey A.* Gomomorfnoe shifrovanie: bezopasnost' oblachnykh vychisleniy i drugie prilozheniya (obzor) [Homomorphic Encryption: Cloud Computing Security and Other Applications (Review)], *Informatika* [Informatics], 2015, No. 1 (45), pp. 90-101.
7. *Babenko L., Burtyka Ph., Makarevich O., Trepacheva A.* Zawiwennye vychisleniya i gomomorfnoe shifrovanie [Secure computing and homomorphic encryption], *III Natsional'nyy superkomp'yuternyy forum (25-27 noyabrya 2014, g. Pereslavl'-Zalesskiy)* [III National Supercomputer Forum (November 25-27, 2014, Pereslavl-Zalessky)]. IPS im. A.K. Aylamazyana RAN, 2014.
8. *Gentry C.* A Fully homomorphic encryption using ideal lattices, *Symposium on the Theory of Computing (STOC).* Bethesda, USA, 2009, pp. 169-178.
9. *Gentry C., Sahai A., Waters B.* Homomorphic Encryption from Learning With Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based // Advances in cryptology – CRYPTO-2013, 33rd Annual Cryptology Conf. Santa Barbara, CA, USA, 2013. Part 1, pp. 73-93.
10. *Parmar P.V.* Survey of various homomorphic encryption algorithms and schemes, *Intern. J. of Computer Applications*, 2014, Vol. 91, No. 8, pp. 26-32.
11. *Jain N., Pal S.K., Upadhyay D.K.* Implementation and analysis of homomorphic encryption schemes, *Intern. J. on Cryptography and Information Security (IJCIS)*, 2012, Vol. 2, No. 2, pp. 27-44.
12. *Smart N., Vercauteren F.* Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes, *Public Key Cryptography – PKC 2010: 13th International Conference on Practice and Theory in Public Key Cryptography, Paris, France, May 26-28, 2010, Proceedings* / P.Q. Nguyen, D. Pointcheval. Berlin, Heidelberg, New York, NY, London [etc.]: Springer Science+Business Media, 2010, pp. 420-443.
13. *Smart N., Vercauteren F.* Fully homomorphic SIMD operations, *Des. Codes Cryptogr*. Springer US, Springer Science+Business Media, 2014, Vol. 71, Iss. 1, pp. 57-81.
14. *Gentry C., Halevi S.* Implementing Gentry's Fully-Homomorphic Encryption Scheme, *Advances in Cryptology – EUROCRYPT 2011: 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011, Proceedings*, K.G. Paterson. Springer Science+Business Media, 2011, pp. 129-148.

15. *Dijk M. v., Gentry C., Halevi S., Vaikuntanathan V.* Fully Homomorphic Encryption over the Integers, *Advances in Cryptology – EUROCRYPT 2010: 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, French Riviera, May 30 - June 3, 2010. Proceedings*, H. Gilbert. – Berlin: Springer Berlin Heidelberg, 2010, pp. 24-43.
16. *Coron J., Mandal A., Naccache D., Tibouchi M.* Fully Homomorphic Encryption over the Integers with Shorter Public Keys, *Advances in Cryptology – CRYPTO 2011: 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011, Proceedings, P. Rogaway*. Springer Science+Business Media, 2011, pp. 487-504.
17. *Burtyka F.B.* Paketnoe simmetrichnoe polnost'yu gomomorfnoe shifrovanie na osnove matrichnykh polinomov [Batch symmetric fully homomorphic encryption based on matrix polynomials], *Tr. Instituta sistemnogo programmirovaniya RAN* [Proceedings of the Institute for System Programming RAS], 2014, Vol. 26. No. 5, pp. 99-116.
18. *Yakovlev M.O.* Zashchishchennyy kal'kulyator. Razrabotka klientskogo komponenta [Secure calculator. Development of the client component], *Vypusknaya kvalifikatsionnaya rabota bakalavra* [Bachelor's final qualification work]. Available at: http://www.nsu.ru/xmlui/bitstream/handle/nsu/471/Text_YakovlevMO.pdf (accessed 20 September 2022).
19. *Babenko L.K., Rusalovskiy I.D.* Metod realizatsii gomomorfnogo deleniya [Homomorphic division implementation method], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2020, No. 4 (214), pp. 212-221.
20. *Babenko L.K., Rusalovskiy I.D.* Masshtabirovanie tsifrovykh izobrazheniy s primeneniem gomomorfnogo shifrovaniya [Digital Image Scaling Using Homomorphic Encryption], *Voprosy kiberbezopasnosti* [Cybersecurity issues], 2021, No. 3 (43), pp. 2-10.

Статью рекомендовал к опубликованию д.т.н., профессор И.А. Калмыков.

Русаловский Илья Дмитриевич – Южный федеральный университет; e-mail: ilya.rusalovskiy@mail.ru; г. Таганрог, Россия; тел.: +79885526701; кафедра безопасности информационных технологий; аспирант.

Бабенко Людмила Клементьевна – e-mail: blk@tsure.ru; тел.: +79054530191; кафедра безопасности информационных технологий; кафедра безопасности информационных технологий; д.т.н.; профессор.

Макаревич Олег Борисович – e-mail: mak@sfedu.ru; тел.: +78634312018; кафедра безопасности информационных технологий; д.т.н.; профессор.

Rusalovsky Il'ya Dmitrievich – Southern Federal University; e-mail: ilya.rusalovskiy@mail.ru; Taganrog, Russia; phone: +79885526701; the department of information technologies security; postgraduate student.

Babenko Lyudmila Kliment'evna – e-mail: blk@tsure.ru; phone: +79054530191; the department of information technologies security; dr. of eng. sc.; professor.

Makarevich Oleg Borisovich – e-mail: mak@sfedu.ru; phone: +78634312018; the department of information technologies security; dr. of eng. sc.; professor.