ТЕХНИЧЕСКИЕ НАУКИ | TECHNICAL SCIENCE

ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ
INFORMATICS, COMPUTER ENGINEERING AND MANAGEMENT

Современная наука и инновации.
2023. № 2(42). С. 10-19
Modern Science and Innovations.
2023; 2(42):10-19

Научная статья / Original article

УДК 004.75
DOI: 10.37493/2307-910X.2023.2.1

Егор Михайлович Ширяев
[Egor M. Shiryaev],
Николай Николаевич Кучеров
[Nikolay N. Kucherov],
Ольга Владимировна Криволапова
[Olga V. Krivolapova]

Разработка алгоритма конфиденциального поиска в распределенных системах

Development of a confidential search algorithm in distributed systems

Северо-Кавказский федеральный университет, г. Ставрополь, Россия /
North-Caucasus Federal University, Stavropol, Russia, eshiriaev@ncfu.ru

Аннотация. *В данной работе рассматриваются распределенные вычислительных системы, а также распределенные системы хранения данных. Учитывая существующие внутренние угрозы для таких систем. В работе предлагается алгоритм конфиденциального поиска для распределенных систем на основе полностью гомоморфного шифрования. Данный алгоритм позволяет производить операцию поиска в узлах распределенной системе не дешифруя информацию в них.*

Ключевые слова: распределенные вычислительные системы, алгоритмы поиска, распределенные хэш-таблицы, гомоморфное шифрование, CKKS

Финансирование: Исследование выполнено за счет гранта Российского научного фонда № 22-71-10046.

Для цитирования: Ширяев Е. М., Кучеров Н. Н., Криволапова О. В. Разработка алгоритма конфиденциального поиска в распределенных системах // *Современная наука и инновации. 2023. № 2(42). С. 10-19. https://doi.org/10.37493/2307-910X.2023.2.1*

Abstract. *This paper deals with distributed computing systems, as well as distributed data storage systems. Given the existing internal threats to such systems. The paper proposes a confidential search algorithm for distributed systems based on fully homomorphic encryption. This algorithm allows you to perform a search operation in the nodes of a distributed system without decrypting the information in them.*

Key words: distributed computing systems, search algorithms, distributed hash tables, homomorphic encryption, CKKS

Funding: The research was supported by Russian Science Foundation Grant No. 22-71-10046.

For citation: Shiryaev E. M., Kucherov N. N. Krivolapova Ol. Vl. Development of a confidential search algorithm in distributed systems // *Modern Science and Innovations. 2023;2(42):10-19. https://doi.org/10.37493/2307-910X.2023.2.1*

Введение. В настоящее время цифровизация большинства сфер деятельности человека породила проблему обработки больших данных [2]. В качестве решения данной проблемы были предложены распределенные вычислительные системы (РВС), а также распределенные системы хранения данных (РСХД). РВС - это системы, в которых вычислительные задачи разбиваются на несколько более мелких задач и распределяются между несколькими узлами (компьютерами, серверами), которые работают параллельно, обмениваясь информацией и результатами вычислений [3]. РСХД - это системы, в которых данные разбиваются на несколько более мелких кусков и хранятся на нескольких узлах, работающих параллельно [1]. РСХД обеспечивают высокую отказоустойчивость и масштабируемость системы, а также обеспечивают более высокую производительность по сравнению с централизованными системами хранения данных. РСХД используются для хранения больших объемов данных, которые не могут быть обработаны на одном компьютере или сервере, а также для обеспечения высокой доступности и надежности данных.

Распределенные системы хранения данных имеют несколько преимуществ по сравнению с централизованными системами:

1. Высокая отказоустойчивость. Распределенные системы хранения данных распределяют данные на несколько узлов, что обеспечивает высокую отказоустойчивость системы. Если один узел выходит из строя, данные могут быть восстановлены из других узлов, что позволяет избежать потери данных.
2. Масштабируемость. Распределенные системы хранения данных могут быть легко масштабированы, добавляя новые узлы к существующей системе. Это позволяет увеличивать производительность системы в зависимости от необходимости, например, при увеличении объема данных.
3. Более высокая производительность. Распределенные системы хранения данных могут выполнять операции чтения и записи данных параллельно на нескольких узлах, что обеспечивает более высокую производительность по сравнению с централизованными системами.
4. Более низкая стоимость. Распределенные системы хранения данных могут быть реализованы на относительно дешевых компьютерах и серверах, что снижает общую стоимость системы. Также, распределенные системы хранения данных позволяют использовать открытые стандарты, что может уменьшить затраты на лицензирование ПО.
5. Лучшая географическая доступность. Распределенные системы хранения данных могут быть развернуты на нескольких узлах, что обеспечивает лучшую географическую доступность к данным. Это может быть особенно полезно для глобальных компаний или для приложений, которые требуют быстрого доступа к данным из разных регионов.

Однако, при всех достоинствах РВС и РСХД они имеют уязвимое место, связанное с безопасностью данных. Таким место являются вычислительные узлы (хранилища данных). Так как в случае предварительного сговора, либо скомпрометированного персонала поставщика таких услуг, злоумышленник может получит прямой доступ к незащищенным данным во время их обработки. Предполагаемым решением является так называемое гомоморфное шифрование (ГШ).

ГШ - это метод криптографии, который позволяет выполнять операции на зашифрованных данных, не расшифровывая их [8]. Это означает, что данные могут оставаться зашифрованными в процессе их обработки, что обеспечивает большую степень безопасности при обработке данных в распределенных вычислительных системах. Вместо передачи открытых данных между узлами, гомоморфное шифрование позволяет зашифровать данные на одном узле и передать зашифрованные данные на другой узел, где они могут быть обработаны, не расшифровывая их. Наиболее эффективным в данном случае является полностью гомоморфное шифрование (ПГШ) [17], так как оно поддерживает, как

гомоморфное сложение, так и гомоморфное умножение. ПГШ позволяет реализовать и другие операции, как это был сделано, например, в работах [11]. Одной из наиболее востребованных операциях в РСХД является операция поиска.

В данной работе ставится задача разработки алгоритма конфиденциального поиска в распределенных системах на основе схем ПГШ. Для решения данной задачи проведено исследование методов сравнения в схемах ПГШ и на основе проведенного анализа предлагается алгоритм поиска данных в зашифрованном виде.

Метод конфиденциальной обработки данных.

1. Полностью гомоморфное шифрование. Идея гомоморфного шифрования была впервые предложена Роном Ривестом, Ади Шамиром и Леонардом Адлеманом, создателями алгоритма RSA [20], в 1978 году. В их исследованиях было установлено, что при шифровании с использованием RSA операция умножения в открытом тексте соответствует операции возведения в степень в зашифрованном тексте. Было предположено, что создание алгоритма шифрования, позволяющего выполнять арифметические операции над зашифрованными данными, позволит повысить эффективность обработки данных. Однако первый конструктивный пример гомоморфного шифрования был предложен только в 2009 году Крейгом Джентри [17]. Эта частично гомоморфная схема шифрования допускала только одну операцию: сложение. Затем, в 2010 году, М. Ван Дейк и др. предложили еще одну частично гомоморфную схему шифрования, которая допускала как аддитивные, так и мультипликативные гомоморфные операции [16].

В 2011 году Крейг Джентри и др. разработали первую полностью гомоморфную схему шифрования, основанную на идее алгоритма шифрования с открытым ключом, в основе которого лежит сложная математическая задача поиска ближайшего вектора [18]. Однако этот алгоритм был не очень эффективен и требовал больших вычислительных ресурсов. В последующие годы были разработаны более эффективные полностью гомоморфные схемы шифрования, такие как Gentry-Halevi и Gentry-Smart, которые уменьшили вычислительную сложность и стали более практичными. Полностью гомоморфное шифрование позволяет выполнять любые комбинации аддитивных и мультипликативных гомоморфных операций, что делает его полезным для выполнения сложных вычислений над зашифрованными данными. Однако, это требует больших вычислительных ресурсов и времени, поэтому на практике используются схемы частично гомоморфного шифрования, которые поддерживают только определенные гомоморфные операции.

В настоящий момент существует множество схем ПГШ. Наиболее интересной для данной работы является схема CKKS, названная по именам авторов Cheon-Kim-Kim-Song, которая обеспечивает приближенное шифрование над рациональными числами с фиксированной точностью).

2. Схема CKKS. CKKS — это схема ПГШ, предложенная в 2016 году. Схема позволяет выполнять вычисления с зашифрованными данными без необходимости их расшифровки, что делает его мощным инструментом для обработки данных с сохранением конфиденциальности. CKKS решает использует приближенную арифметику [10]. Это позволяет выполнять вычисления ПГШ с меньшей точностью, что приводит к сокращению времени вычислений.

CKKS также основана на задаче обучение с ошибками в кольце (ООК). Схема работает путем шифрования данных в полиномы с комплексными коэффициентами, которые затем обрабатываются с использованием методов на основе ООК [21]. Приближенная арифметика используется для управления ошибками, возникающими во время вычислений. То есть схема может быть использована для арифметических операций над полем $C^{N/2}$. Пространство открытого текста и пространство зашифрованного текста занимают одну и ту же область

$$Z_Q\left[X\right]/\left(X^N+1\right) \tag{1}$$

где N – степень двойки.

Пакетное кодирование этой схемы $C^{\frac{N}{2}} \leftrightarrow Z_Q[X]/(X^N+1)$ отображает массив комплексных чисел в многочлен со свойством: $decode(encode(m_1) \otimes encode(m_2)) \approx m_1 \square m_2$, где $\otimes$ является компонентным умножением, и $\square$ является нециклической сверткой. Схема CKKS поддерживает стандартные рекомендуемые параметры, выбранные для обеспечения 128-битной безопасности для единого троичного закрытого ключа. $s \in_u \{-1,0,1\}^N$, по группе гомоморфных стандартов шифрования. CKKS кодирует поле комплексных чисел с помощью полиномов Лагранжа [5]. Схема CKKS имеет большой потенциал для развития, так как ее арифметика выполняется на поле рациональных чисел с фиксированной точностью. Рассмотрим более подробно гомоморфные арифметические операции в CKKS.

Шифрование: чтобы зашифровать сообщение с использованием схемы CKKS, текстовое сообщение сначала кодируется как вектор комплексных чисел. Затем из этого вектора генерируется многочлен открытого текста. Для заданного многочлена $m \in \mathrm{R}$ вывод зашифрованного текста $c \in \mathrm{R}_{qL}^h$. Шифрование c для m будет удовлетворять условию $c, sk = m + e(mod\, q_L)$ для небольшого e. Постоянный $Bclean$ точка ограничения возможностей шифрования, т. е. полином ошибок нового зашифрованного текста удовлетворяет $|e|_\infty^{can} \le Bclean$ с высокой вероятностью.

Расшифровка: для расшифровки зашифрованного текста в схеме CKKS используется специальный ключ, известный как секретный ключ. Генерация секретного значения sk, публичная информация pk для шифрования и оценочного ключа evk. Для зашифрованного текста c на уровне выходного полинома $m \leftarrow c, sk(mod\, q_l)$ для секретного ключа sk.

Сложение: чтобы выполнить сложение двух зашифрованных текстов в схеме CKKS, их соответствующие полиномы добавляются по модулю q. Результатом является новый полином, который можно расшифровать, чтобы получить сумму исходных сообщений открытого текста. Для заданных шифров m_1 и m_2 выходное шифрование $m_1 + m_2$. Ошибка выходного зашифрованного текста ограничена суммой двух ошибок во входных зашифрованных текстах.

Умножение: Умножение в схеме CKKS более сложное. Чтобы умножить два зашифрованных текста, сначала перемножаются их соответствующие полиномы открытого текста. Полученный полином имеет коэффициенты, кратные q^2. Для пары зашифрованных текстов (c_1, c_2) вывести зашифрованный текст $c_{mult} \in \mathrm{R}_{ql}^h$, удовлетворяющие условию $c_{mult}, sk = c_1, sk \cdot c_2, sk + e_{mult}(modq_l)$ для некоторого многочлена $\$e_{mult} \in \mathrm{R}$ с $|e_{mult}|^{can_\infty} B_{mult}(l)$.

Релинеализация: для зашифрованного текста $\boldsymbol{c} \in \mathrm{R}_{ql}^h$ на уровне l и нижний уровень $l' < l$, выходной зашифрованный текст $\boldsymbol{c}' \leftarrow \frac{q_{l'}}{q_l}$ в R_{ql}^h т.е., $\boldsymbol{c}'$ получен масштабированием $\frac{q_{l'}}{q_l}$ к элементам $\boldsymbol{c}$ и округление коэффициентов до целых чисел. Потом индекс падает $l \to l'$ когда $l' = l - 1$.

Схема CKKS также поддерживает полиноминальные операции. С помощью данных операций возможно разработать операцию сравнения зашифрованных секций, что будет рассмотрено в следующем разделе.

Сравнение в полностью гомоморфном шифровании. Для реализации сравнения возможно применение приближенной функции знака. С одной стороны функция определения знака числа не является непрерывной на множестве $[-1,1]$, то в окрестности точки неустранимого разрыва первого рода $x = 0$ находится один из локальных максимумов

ошибки аппроксимации многочленом, с другой стороны для уменьшения погрешности аппроксимации необходимо увеличивать степень многочлена аппроксимации, что приводит к увеличению количества умножений и увеличению мультипликативной глубины [9].

Для уменьшения вычислительной сложности было предложено использование метода Ньютона для алгоритма нахождения квадратного корня числа, который позволяет вычислить приблизительное значение функции квадратного корня итеративным методом, а также несколько его модификаций для улучшения сходимости к функции знака [13, 22], но в этом методе требуется операция деления и ее полиномиальная аппроксимация, которая не реализована в ПГШ, либо реализована не эффективно. Также был разработан метод Ньютона-Шульца [19], который отличается тем, что использует отличную от предыдущего подхода функцию аппроксимации операции деления.

Алгоритмы сравнения в схемах ПГШ исходят из того, что сложение и умножение являются основными поддерживающимися операциями. Суть метода, предложенного в [12], заключается аппроксимации функции знака композиций полиномов $f\circ\ldots\circ f\circ g\circ\ldots\circ g$, причем эти полиномы должны быть выбраны так, чтобы для достижения хорошей точности требовалось небольшого количества композиций. Рассмотрим данную функцию.

Функция $f(x)$ удовлетворяет следующим свойствам:

Нечетная функция:

1. $f(1)=1, f(-1)=-1$
2. $f'(x)=c(1-x)^n(1+x)^n$ для некоторой константы $c>0$

Таким образом: $f(x)$:

$$f_n(x)=\sum_{i=0}^{n}\frac{1}{4^i}\cdot\binom{2i}{i}\cdot x\left(1-x^2\right)^i \qquad (2)$$

С ее помощью были вычислены: $f_n(x)$:

- $f_1(x)=-\frac{1}{2}x^3+\frac{3}{2}x$
- $f_2(x)=\frac{3}{8}x^5-\frac{10}{8}x^3+\frac{15}{8}x$
- $f_3(x)=-\frac{5}{16}x^7+\frac{21}{16}x^5-\frac{35}{16}x^3+\frac{35}{16}x$
- $f_4(x)=\frac{35}{128}x^9-\frac{180}{128}x^7+\frac{378}{128}x^5-\frac{420}{128}x^3+\frac{315}{128}x$

Для ускорения используют также многочлен $g(x)$, который удовлетворяют следующим свойствам свойствами:

Нечетная функция:

$\exists 0<\delta<1 \text{s.t.} x<g(x)\le 1$ для всех $x\in(0,\delta], u\, g([\delta,1])\subseteq[1-\tau,1]$

Примерами $g(x)$ являются:

- $g_1(x)=-\frac{1359}{2^{10}}\cdot x^3+\frac{2126}{2^{10}}\cdot x$
- $g_2(x)=\frac{3796}{2^{10}}\cdot x^5-\frac{6108}{2^{10}}\cdot x^3+\frac{3334}{2^{10}}\cdot x$
- $g_3(x)=-\frac{12860}{2^{10}}\cdot x^7+\frac{25614}{2^{10}}\cdot x^5-\frac{16577}{2^{10}}\cdot x^3+\frac{4589}{2^{10}}\cdot x$
- $g_4(x)=\frac{46623}{2^{10}}\cdot x^9-\frac{113492}{2^{10}}\cdot x^7+\frac{97015}{2^{10}}\cdot x^5-\frac{34974}{2^{10}}\cdot x^3+\frac{5850}{2^{10}}\cdot x$

Оценка эффективности и точности этих полиномов представлена в [7]. Таким

образом, на основе данных полиномов можно получить приближенное значение знака рассматриваемого шифр текста. Это позволяет реализовать сравнение в схеме ПГШ.

Поиск информации в распределённых системах. Алгоритмы поиска в распределенных системах относятся к методам и подходам, используемым для обнаружения информации, данных или ресурсов в сети взаимосвязанных узлов, которые могут быть географически распределены. Эти алгоритмы нацелены на эффективный поиск необходимой информации при минимизации сетевого трафика и задержки. Рассмотрим наиболее популярные методы.

1. Распределенные хэш-таблицы (РХТ) — это метод распределенных вычислений, используемый для индексации и извлечения данных в крупномасштабной распределенной системе [14]. Они особенно полезны для одноранговых сетей, сетей распространения контента и других децентрализованных систем, в которых данные распределяются по нескольким узлам.
2. Итеративный поиск в глубину (ИПГ) — это алгоритм поиска, который сочетает в себе преимущества поиска в глубину (ПГ) и поиска в ширину (ПШ), избегая при этом их ограничений [23]. Это особенно полезно для поиска в больших деревьях или графах в среде с ограниченным объемом памяти. Основная идея ИПГ заключается в выполнении серии поисков ПГ с увеличением пределов глубины до тех пор, пока не будет найден нужный узел. На каждой итерации поиск выполняется на ограниченном наборе узлов, что делает его более эффективным, чем полный поиск ПШ.
3. Случайное блуждание: при поиске случайного блуждания запрос отправляется на случайный узел в сети, и узел пересылает запрос случайно выбранному соседу [6]. Этот процесс продолжается до тех пор, пока не будут найдены запрошенные данные. Алгоритмы случайного блуждания используются в крупномасштабных сетях, где топология сети часто меняется.
4. Поиск на основе градиента: при поиске на основе градиента запрос перенаправляется на узел, ближайший к месту назначения, на основе некоторой метрики, такой как сетевая задержка или сетевое расстояние [15]. Этот алгоритм уменьшает сетевой трафик за счет минимизации количества узлов, которые необходимо обыскать.

Рассмотрим метод РХТ подробнее. РХТ — это распределенная система, которая организует набор узлов (или одноранговых узлов) в виртуальную оверлейную сеть. Каждому узлу присваивается уникальный идентификатор, обычно хеш-значение.

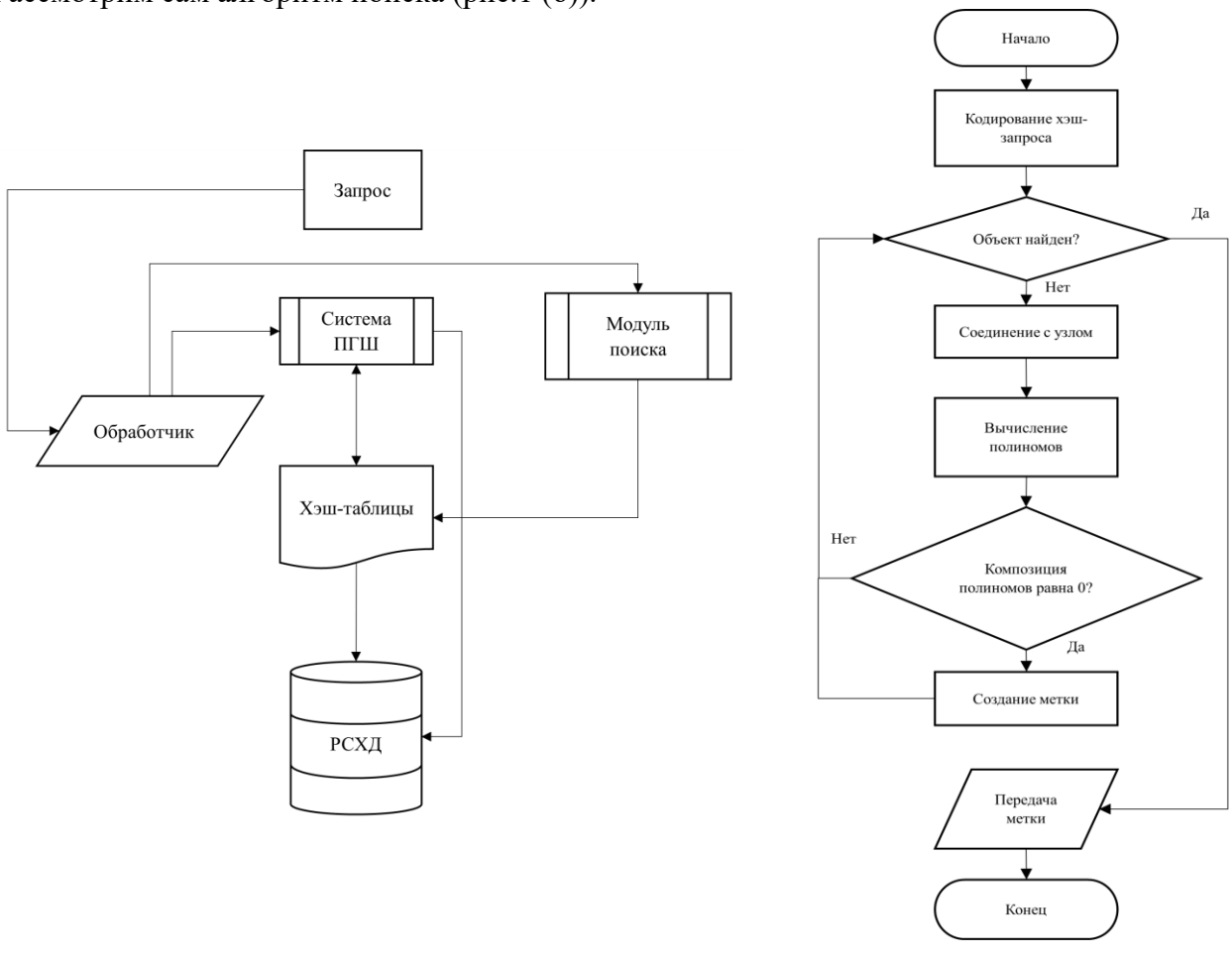
Алгоритм РХТ обеспечивает равномерное распределение узлов по сети, и каждый узел поддерживает таблицу маршрутизации, которая сопоставляет подмножество ключей с соседними узлами. Когда узел хочет найти данные с определенным ключом, он сначала хэширует ключ для создания идентификатора. Затем он запрашивает узел с идентификатором, наиболее близким к целевому идентификатору в таблице маршрутизации. Этот узел может не иметь запрошенных данных, но он может предоставить идентификатор следующего ближайшего узла в сети, и процесс продолжается до тех пор, пока не будет найден узел с запрошенными данными.

РХТ имеют несколько преимуществ по сравнению с другими методами распределенного индексирования и поиска, такими как лавинная рассылка и итеративное углубление. Они обеспечивают эффективное и масштабируемое извлечение данных даже в крупномасштабных сетях с динамической топологией. Они также обеспечивают отказоустойчивость и устойчивость к сбоям узлов, поскольку данные реплицируются между несколькими узлами в сети.

Для данной работы наибольший интерес представляет метод РХТ, так как в случае ГШ обработка и поиск хэш-таблиц является более вычислительно простой задачей, чем поиск и анализ узлов вслепую или по очереди. Метод РЧТ имеет недостаток, связанный с нагрузкой на маршрутизаторы, что возможно компенсировать, применяя дополнительные маршрутизаторы, так и применяя методы балансировки нагрузки, либо особую архитектуру РВС и РСХД.

Разработка алгоритма. В данной работе представляется алгоритм конфиденциального поиска. За основу алгоритма предлагается взять метод PXT. Так как наличие хеш-таблиц позволяет сократить вычислительные затраты при поиске. В ином случае необходимо сравнивать информацию на каждом узле. Так как рассмотренная операция сравнения в CKKS все же является более вычислительно сложным, чем обычное сравнение эффективнее будет сравнивать данные из хеш-таблиц и поиск информации уже на точно определенном узле. Особенность предлагаемого алгоритма заключается в применении схем ПГШ. Общая блок-схема работы такой системы представлена на рис.1 (а)

Таким образом, РСХД защищена от доступа извне. Обработка данных осуществляется через модуль хэш-таблиц. Хеш-таблица, как и данные в РСХД зашифрованы. Кроме того, при обращении к таблице, либо к элементам РСЧД не требуется дешифрование данных, находящихся в них, так как ПГШ позволяет обрабатывать данные конфиденциально. Рассмотрим сам алгоритм поиска (рис.1 (б)).



а б

Рисунок 1. (а) Схема системы, построенной на основе алгоритма; (б) Блок-схема алгоритма
Figure 1. (a) The scheme of the system based on the algorithm; (b) Block diagram of the algorithm

Рассмотрим положительные и отрицательные свойства полученного алгоритма. Взаимодействие с РСХД происходит на основе запроса метки данных находящихся на определенном узле РСХД, т. е. пользователь не взаимодействует с данными напрямую, доступ к данным осуществляется только на основе полученной метки при необходимом уровне авторизации. Это позволяет обеспечить РСХД от хищения информации при несанкционированном доступе, например, при компрометации персонала поставщика услуг. Даже если злоумышленник получит информацию с РСХД она не будет имеет смысла, так как будет зашифрована. Требуемый для расшифровки закрытый ключ имеется в доступе только у авторизированного пользователя. Таким образом полученный алгоритм обеспечивает

необходимый уровень безопасности с возможностью взаимодействия системой. Это может быть полезно для медицинских, административных, банковских и других систем, которые чувствительны к краже информации.

Из недостатков можно вынести следующее. Система чувствительна к DDOS-атакам [4]. Так как даже получая ложные запросы, система все равно обрабатывает их (недостаток РХТ систем). Несмотря на преимущества для РСХД систем, для РВС данная система будет иметь низкую производительность, так как в РВС предполагается постоянное изменение, перемещение и стирание данных, а также их миграция по узлам сети, что накладывает нагрузку как на РХТ, так и повышает объем вычислений в ПГШ.

Данные недостатки будут исследованы в будущих работах. Предполагается повышение противостояние DDOS-атакам, путем изменения системы обработки запроса. Для РВС предполагаются исследования, направленные на оптимизацию вычислительной нагрузки, связанной с ПГШ. Кроме того, планируются практические исследования полученного алгоритма в виде программной реализации.

Заключение. В данной работе было проведено исследования РСВ. Акцент ставился на повышение конфиденциальности РСХД, путем создания алгоритма конфиденциального поиска. Для выполнения поставленной задачи был определен метод конфиденциальной обработки данных. Метод представлен в виде схемы ПГШ CKKS, которая позволяет обрабатывать в зашифрованном виде рациональные числа с фиксированной точностью. Кроме того, были проведены исследования методов поиска в РСХД. В качестве наиболее подходящего к модификации был определен метод РХТ, который производит поиск на основе таблиц хе-функций. Был реализован алгоритм конфиденциального поиска на основе функции сравнения в схеме CKKS. Функция сравнения реализуется на основе вычисления композиции полиномов.

Таким образом был получен алгоритм конфиденциального поиска. В качестве дальнейших исследований планируется программная реализация алгоритма, а также устранение его недостатков для расширения возможностей его применения.

ЛИТЕРАТУРА

1. Мазур Э. М. Распределенные системы хранения данных: анализ, классификация и выбор // Перспективы развития информационных технологий. 2015. № 26. С. 33–60.

2. Марц Н., Уоррен Д. Большие данные: принципы и практика построения масштабируемых систем обработки данных в реальном времени. М.: Вильямс. 2017. 368 с.

3. Радченко Г. И. Распределенные вычислительные системы // Челябинск: Фотохудожник. 2012. С. 184.

4. Харитонов В. С., Черяпкин Д. П. DDoS-атака: классификация и особенности // Постулат. 2016. № 12 (14). С. 45.

5. Шакиров И. А. О фундаментальных характеристиках семейства интерполяционных полиномов Лагранжа // Известия Саратовского университета. Новая серия. Серия Математика. Механика. Информатика. 2013. № 1–2 (13). С. 99–104.

6. Akyildiz I. F. [et al.]. A new random walk model for PCS networks // IEEE Journal on selected areas in communications. 2000. No. 7 (18). P. 1254–1260.

7. Babenko M. [et al.]. Experimental Evaluation of Homomorphic Comparison Methods // Proceedings - 2020 Ivannikov Ispras Open Conference, ISPRAS 2020. 2020. P. 69–74.

8. Chauhan K. K., Sanger A. K., Verma A. Homomorphic encryption for data security in cloud computing // International Conference on Information Technology (ICIT), Bhubaneswar, India. 2015. P. 206–209.

9. Chen H., Chillotti I., Song Y. Improved bootstrapping for approximate homomorphic encryption // Advances in Cryptology – EUROCRYPT 2019, Y. Ishai and V. Rijmen, Eds. Cham: Springer International Publishing, 2019, P. 34–54.

10. Cheon J. H. [et al.]. Homomorphic encryption for arithmetic of approximate numbers // International Conference on the Theory and Application of Cryptology and Information Security. Springer, 2017. P. 409–437.

11. Cheon J. H., Kim A. Homomorphic encryption for approximate matrix arithmetic // Cryptology ePrint Archive. 2018.

12. Cheon J. H., Kim D., Kim D. Efficient homomorphic comparison methods with optimal complexity // Proceedings of 2020 International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2020). Springer, 2020. P. 221–256.

13. Cordero A. [et al.]. Numerically stable improved Chebyshev–Halley type schemes for matrix sign function // Journal of Computational and Applied Mathematics. 2017. No. 318. P. 189–198.

14. Dabek F. F. E. A distributed hash table 2005.

15. Darken C., Moody J. Towards faster stochastic gradient search // Advances in neural information processing systems. 1991. Vol. 4.

16. Dijk M. van [et al.]. Fully homomorphic encryption over the integers // EUROCRYPT 2010, LNCS, Springer. 2010. Vol. 6110. P. 24–43.

17. Gentry C. A fully homomorphic encryption scheme / C. Gentry, Stanford university, 2009.

18. Gentry C., Halevi S. Implementing gentry's fully-homomorphic encryption scheme // EUROCRYPT 2011. LNCS. Springer, 2011. Vol. 6632. P. 129–148.

19. Higham N. J. Functions of matrices: theory and computation / N. J. Higham. SIAM, 2008.

20. Kohnfelder L. M. Towards a practical public-key cryptosystem. 1978.

21. Lyubashevsky V., Peikert C., Regev O. On ideal lattices and learning with errors over rings // Journal of the ACM (JACM). 2013. No. 6 (60). P. 1–35.

22. Nakatsukasa Y., Bai Z., Gygi F. Optimizing Halley's iteration for computing the matrix polar decomposition // SIAM Journal on Matrix Analysis and Applications. 2010. No. 5 (31). P. 2700–2720.

23. Pereira R. F. [et al.]. Iterative Depth-First Search for FOND Planning 2022. P. 90–99.

REFERENCES

1. Mazur EH. M. Raspredelennye sistemy khraneniya dannykh: analiz, klassifikatsiya i vybor // Perspektivy razvitiya informatsionnykh tekhnologii. 2015. No. 26. P. 33-60.

2. Marts N., Uorren D. Bol'shie dannye: Printsipy i praktika postroeniya masshtabiruemykh sistem obrabotki dannykh v real'nom vremeni. M.: Vil'yams. 2017. 368 p.

3. Radchenko G. I. Raspredelennye vychislitel'nye sistemy // Chelyabinsk: Fotokhudozhnik. 2012. P. 184.

4. Kharitonov V. S., Cheryapkin D. P. DDoS-ataka: klassifikatsiya i osobennosti // Postulat. 2016. № 12 (14). P. 45.

5. Shakirov I. A. O fundamental'nykh kharakteristikakh semeistva interpolyatsionnykh polinomov Lagranzha // Izvestiya Saratovskogo universiteta. Novaya seriya. Seriya Matematika. Mekhanika. Informatika. 2013. No. 1-2 (13). P. 99-104.

6. Akyildiz I. F. [et al.]. A new random walk model for PCS networks // IEEE Journal on selected areas in communications. 2000. No. 7 (18). P. 1254–1260.

7. Babenko M. [et al.]. Experimental Evaluation of Homomorphic Comparison Methods // Proceedings - 2020 Ivannikov Ispras Open Conference, ISPRAS 2020. 2020. P. 69–74.

8. Chauhan K. K., Sanger A. K., Verma A. Homomorphic encryption for data security in cloud computing // International Conference on Information Technology (ICIT), Bhubaneswar, India. 2015. P. 206–209.

9. Chen H., Chillotti I., Song Y. Improved bootstrapping for approximate homomorphic encryption // Advances in Cryptology – EUROCRYPT 2019, Y. Ishai and V. Rijmen, Eds. Cham: Springer International Publishing, 2019, P. 34–54.

10. Cheon J. H. [et al.]. Homomorphic encryption for arithmetic of approximate numbers // International Conference on the Theory and Application of Cryptology and Information Security. Springer, 2017. P. 409–437.

11. Cheon J. H., Kim A. Homomorphic encryption for approximate matrix arithmetic // Cryptology ePrint Archive. 2018.

12. Cheon J. H., Kim D., Kim D. Efficient homomorphic comparison methods with optimal complexity // Proceedings of 2020 International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2020). Springer, 2020. P. 221–256.

13. Cordero A. [et al.]. Numerically stable improved Chebyshev–Halley type schemes for matrix sign function // Journal of Computational and Applied Mathematics. 2017. No. 318. P. 189–198.

14. Dabek F. F. E. A distributed hash table 2005.

15. Darken C., Moody J. Towards faster stochastic gradient search // Advances in neural information processing systems. 1991. Vol. 4.

16. Dijk M. van [et al.]. Fully homomorphic encryption over the integers // EUROCRYPT 2010, LNCS, Springer. 2010. Vol. 6110. P. 24–43.

17. Gentry C. A fully homomorphic encryption scheme / C. Gentry, Stanford university, 2009.

18. Gentry C., Halevi S. Implementing gentry's fully-homomorphic encryption scheme // EUROCRYPT 2011. LNCS. Springer, 2011. Vol. 6632. P. 129–148.

19. Higham N. J. Functions of matrices: theory and computation / N. J. Higham. SIAM, 2008.

20. Kohnfelder L. M. Towards a practical public-key cryptosystem. 1978.

21. Lyubashevsky V., Peikert C., Regev O. On ideal lattices and learning with errors over rings // Journal of the ACM (JACM). 2013. No. 6 (60). P. 1–35.

22. Nakatsukasa Y., Bai Z., Gygi F. Optimizing Halley's iteration for computing the matrix polar decomposition // SIAM Journal on Matrix Analysis and Applications. 2010. No. 5 (31). P. 2700–2720.

23. Pereira R. F. [et al.]. Iterative Depth-First Search for FOND Planning 2022. P. 90–99.

ОБ АВТОРАХ / ABOUT THE AUTHORS

Ширяев Егор Михайлович – аспирант кафедры вычислительной математики и кибернетики факультета математики и компьютерных наук имени профессора Н.И. Червякова ФГАОУ ВО «Северо-Кавказский федеральный университет». 355000, Россия, г. Ставрополь, тел. 8-960-424-33-09, e-mail: eshiriaev@ncfu.ru

Shiryaev Egor Mikhailovich – Postgraduate Student, Department of Computational Mathematics and Cybernetics, Faculty of Mathematics and Computer Science named after Professor N.I. Chervyakov FSAEI HE "North-Caucasus Federal University". 355000, Russia, Stavropol, tel. 8-960-424-33-09, e-mail: eshiriaev@ncfu.ru

Кучеров Николай Николаевич – старший научный сотрудник учебно-научного центра «Вычислительной математики и параллельного программирования на суперЭВМ» факультета математики и компьютерных наук имени профессора Н.И. Червякова ФГАОУ ВО «Северо-Кавказский федеральный университет». 355000, Россия, г. Ставрополь, тел. 8-865-295-68-00, e-mail: nkucherov@ncfu.ru

Kucherov Nikolay Nikolaevich – Senior Researcher Educational and Scientific Center "Computational Mathematics and Parallel Programming on Supercomputers" Faculty of Mathematics and Computer Science named after Professor N.I. Chervyakov FSAEI HE "North-Caucasus Federal University"., 355000, Russia, Stavropol, 355000, tel. 8-865-295-68-00, E-mail: nkucherov@ncfu.ru

Криволапова Ольга Владимировна – аспирант кафедры вычислительной математики и кибернетики факультета математики и компьютерных наук имени профессора Н.И. Червякова ФГАОУ ВО «Северо-Кавказский федеральный университет». 355000, Россия, г. Ставрополь, тел. 8-928-639-33-51, e-mail: obabchenko@ncfu.ru

Krivolapova Olga Vladimirovna – Postgraduate Student, Department of Computational Mathematics and Cybernetics, Faculty of Mathematics and Computer Science named after Professor N.I. Chervyakov FSAEI HE "North-Caucasus Federal University". 355000, Russia, Stavropol, tel. 8-928-639-33-51, e-mail: obabchenko@ncfu.ru

Дата поступления в редакцию: 12.04.2023
После рецензирования:13.05.2023
Дата принятия к публикации:07.06.2023