Андрей В. Горлатых, Сергей В. Запечников
Национальный исследовательский ядерный университет «МИФИ»,
Каширское ш., 31, г. Москва, 115409, Россия
e-mail: grand61rus@gmail.com, http://orcid.org/0000-0003-0462-3339
e-mail: svzapechnikov@mephi.ru, http://orcid.org/0000-0002-7975-6040

ПОСТРОЕНИЕ ЗАЩИЩЕННОЙ СИСТЕМЫ УПРАВЛЕНИЯ МНОГОМЕРНЫМИ СТРУКТУРАМИ ДАННЫХ

DOI: http://dx.doi.org/10.26583/bit.2018.3.02

Аннотация. В настоящее время нельзя недооценивать значимость информационных технологий во всех сферах человеческой жизни. Вычислительные технологии и средства глубоко проникли во все виды деятельности человека и стали их неотъемлемой частью. Однако с учетом того, что информация стала стратегически важным ресурсом, на первый план вышла необходимость обеспечения ее безопасности. Большое количество современных исследований посвящено обеспечению информационной безопасности вычислительных технологий. В последнее время стали набирать популярность технологии, основанные на использовании многомерных структур данных. Это в свою очередь привело к возникновению противоречия, заключающегося в том, что растущий спрос на информационные технологии, основанные на многомерных структурах данных, не подкреплен современными исследованиями в области информационной безопасности подобных структур. Сложилась ситуация, при которой обработка конфиденциальной информации, построенной по многомерному принципу, является невозможной в силу отсутствия систем управления многомерными структурами данных, обеспечивающими надлежащий уровень защиты. Авторами было принято решение о разработке подобной системы. В статье предложена архитектура системы управления многомерными структурами данных, позволяющая обеспечить безопасное управление информацией, построенной по многомерному принципу. Архитектура включает в себя описание основных компонентов, входящих в систему, задач, выполняемых этими компонентами, а также принципы организации взаимодействия между ними. В статье описывается способ построения системы управления многомерными структурами данных, которая позволяет осуществлять безопасное чтение или запись конфиденциальной информации, представленной в виде многомерных кубов данных. Предлагаемое решение позволяет обеспечить безопасную обработку конфиденциальной многомерной информации, расширяя тем самым область применения подобных систем.

Ключевые слова: хранилища документов, поисковые запросы, криптографические протоколы, аутентификация, электронная подпись.
Для цитирования: ГОРЛАТЫХ, Андрей В.; ЗАПЕЧНИКОВ, Сергей В.. ПОСТРОЕНИЕ ЗАЩИЩЕННОЙ СИСТЕМЫ УПРАВЛЕНИЯ МНОГОМЕРНЫМИ СТРУКТУРАМИ ДАННЫХ. Безопасность информационных технологий, [S.l.], n. 3, p. 16-25, 2018. ISSN 2074-7136. Доступно на: <https://bit.mephi.ru/index.php/bit/article/view/1136>. Дата доступа: 28 aug. 2018. doi:http://dx.doi.org/10.26583/bit.2018.3.02.

Andrey V. Gorlatyh, Sergey V. Zapechnikov
National Research Nuclear University MEPhI (Moscow Engineering Physics Institute),
Kashirskoye shosse 31 Moscow, 115409, Russia
e-mail: grand61rus@gmail.com, http://orcid.org/0000-0003-0462-3339
e-mail: svzapechnikov@mephi.ru, http://orcid.org/ 0000-0002-7975-6040

Building secure multidimensional data management system

DOI: http://dx.doi.org/10.26583/bit.2018.3.02

Abstract. Nowadays one should not underestimate importance of informational technologies across all fields of humans' life. Computational technologies and tools became crucial parts of any human activity. However, since information is becoming a strategic resource, it is necessary to ensure its security by all means. A large number of contemporary studies have been carried on in the field of informational security. Recently, multidimensional information technologies have become popular for data analysis and processing. This has led to a contradiction between ever growing demand of multidimensional data processing systems and inability to properly secure such system due to insufficient research within this field. As a result, sensitive multidimensional data cannot be securely processed due to lack of proper

secure systems able to ensure appropriate level of informational security. Authors decided to develop such system. The paper proposes architecture of multidimensional data management system capable of secure information management built up as a multidimensional hypercube. Proposed architecture consists of a description of main system components, a description of task performed by those components and principles of their communication. The paper describes a way to build a system which ensures safe read/write operations for confidential information organized as multidimensional hypercube. The proposed solution allows secure processing of sensitive multidimensional data and therefore extending the area where such systems can be used.

Keywords: data warehouses, search queries, cryptographic protocols, authentication, multidimensional data.

For citation: GORLATYH, Andrey V.; ZAPECHNIKOV, Sergey V.. Building secure multidimensional data management system.IT Security (Russia), [S.l.], n. 3, p. 16-25, 2018. ISSN 2074-7136. Available at: <https://bit.mephi.ru/index.php/bit/article/view/1136>. Date accessed: 28 aug. 2018. doi:http://dx.doi.org/10.26583/bit.2018.3.02

Введение

Информационные технологии все больше проникают в повседневную жизнь. Структуры бизнеса – всевозможные корпорации, агропромышленные холдинги, торговые компании, негосударственные фонды, коммерческие партнерства давно и плодотворно используют достижения IT технологий, такие как СУБД различных производителей, для автоматизации своего бизнеса.

В последнее время в этот процесс вовлекается и государственный сектор [1,2]. Сегодня электронный документооборот и электронная подпись стали повсеместной нормой. Госуслуги и работа в режиме одного окна также прочно вошли в нашу жизнь. Практически каждое ведомство или министерство РФ ведет свои госреестры и собирает информацию по своему профилю в рамках обозначенной деятельности. Так, в настоящее время, Министерство налогов и сборов полностью автоматизировало процедуру сбора налоговой отчетности и фискальных проверок, сбора информации по кассовым операциям в режиме онлайн. Автоматизирована процедура госзакупок и гособоронзаказа посредством ведения всеми участниками ЕИС (единых информационных систем). Существующие ранее платежные системы и введенная новая платежная система «Мир» в совокупности с СУБД, учитывающими денежные потоки внутри банков, также постоянно модифицируются и обновляются. Государственные фонды, ведущие учет социального обеспечения граждан, такие как Пенсионный фонд РФ, Фонд социального страхования, Фонд медицинского страхования также ведут свои базы данных и соответствующие реестры.

Наиболее распространенные на сегодняшний день реляционные СУБД, предназначенные в основном для учета транзакций, сбора и учета информации в реальном масштабе времени, наряду с определенными преимуществами имеют и ряд недостатков, или, точнее сказать, ограничений в силу того, что реляционный подход Кодда сам по себе является ограничением с большим количеством условий.

В сравнении с многомерными массивами данных требование быстрой агрегации - основная вычислительная проблема для реляционных БД [3]. Для ускорения агрегации данные в них, как правило, слишком денормализованы, т.е. хранятся не очень эффективно с точки зрения занимаемого места на диске и контроля целостности БД, а также дополнительно содержат вспомогательные таблицы, хранящие частично агрегированные данные. Для того, чтобы достичь достаточного уровня производительности, для реляционных систем требуется специальная настройка и особые способы индексации данных. Ограничения SQL не позволяют реализовать в реляционных СУБД многие встроенные функции, легко обеспечиваемые в системах, основанных на многомерном представлении данных.

Однако одновременно с развитием технологий обработки многомерных структур данных растет и число информационных угроз. Как и в любой другой информационной отрасли, здесь остро встает вопрос обеспечения безопасности информации,

организованной по многомерному принципу. В настоящее время не существует систем, которые бы могли не только полностью обеспечить безопасность обрабатываемой информации, но и показать высокий уровень производительности. Зачастую предлагаемое решение является либо малопроизводительным, либо обеспечивает защиту информации лишь для узкого круга задач [4].

Кроме того, отсутствие эффективных средств комплексной защиты информации для многомерных массивов данных тормозит широкое распространение многомерных структур в различные сферы жизни человека. Отсюда вытекает необходимость разрешить противоречие в том, что при повышении заинтересованности к многомерным системам и повышении ценности информации, хранящейся в них, нет эффективных средств защиты этой информации.

В части безопасной обработки зашифрованных конфиденциальных данных в качестве проектов, которыми вдохновлялся исследователь данной работы, можно выделить проекты, обеспечивающие информационную безопасность реляционных баз данных, содержащих конфиденциальную информацию:

- CryptDB [5,6];
- Arx [7];
- Oblivious Transfer with Access Control [8,9].

Однако проблема данных проектов заключается в том, что они не учитывают специфику построения многомерных структур данных, а значит они не могут быть легко перенесены на случай обработки конфиденциальной информации, построенной по многомерному принципу.

Это приводит к тому, что появляется необходимость самостоятельно синтезировать решение, которое бы обеспечило надежное управление многомерными массивами данных, обеспечивая при этом надлежащий уровень безопасности информации в системе.

В качестве механизма разграничения прав доступа используется разграничение прав доступа на основе атрибутов. При этом для обеспечения конфиденциальности информации одновременно с разделением прав доступа на основании атрибутов используются схемы атрибутного шифрования [10-13].

В настоящей статье рассматривается архитектура системы, позволяющей безопасную обработку конфиденциальной информации, построенной по многомерному принципу.

Оставшаяся часть статьи организована следующим образом. В п. 1 рассматриваются требования, предъявляемые к системе. Далее, в п. 2 – общее описание архитектуры и входящих в нее компонентов. Далее, в п. 3, 4, 5 и 6, более подробно описывается каждый из компонентов системы, а именно Сервис Атрибутов, Прокси-Сервис, Сервис Контроля Атрибутов и Хранилище данных соответственно. В «Заключении» подводятся итоги и формулируются основные результаты работы.

1. Требования, предъявляемые к системе

При проектировании защищенной системы обработки многомерных информационных массивов необходимо специфицировать требования, которые предъявляются к системам подобного класса. Наличие такого списка требований позволит на заключительном этапе проектирования убедиться в том, что архитектура предложенной системы соответствует всем заявленным требованиям. В рамках работы выделяется два набора требований: функциональные требования и требования безопасности.

Функциональные требования. Данные требования описывают набор функций, которыми должна обладать система, осуществляющая управление многомерными структурами данных. К требованиям относят:

1) механизм управления учетными записями пользователей: система должна обладать функциями по регистрации, хранению и удалению учетных записей пользователей, зарегистрированных в системе;

2) механизм обработки запросов: система должна реализовывать функции, позволяющие корректно получить, проанализировать и выполнить пользовательские запросы на чтение/запись информации;

3) наличие у каждого компонента механизмов хранения служебной информации: каждый компонент системы в процессе выполнения пользовательских запросов пользуется служебной информацией о системах или данных. Необходимо, чтобы каждый компонент системы обладал возможностью хранения подобной служебной информации в выделенной БД;

4) механизм взаимодействия с программным интерфейсом сторонних хранилищ данных: необходимо обеспечить функции чтения/записи информации в хранилище данных путем выполнения запросов к программному интерфейсу хранилища, предоставляемому поставщиком услуг.

Требования безопасности. Здесь описываются требования, предъявляемые к системе управления многомерными данными для того, чтобы она обеспечивала надлежащий уровень безопасности данных, хранящихся в системе. Среди основных требований безопасности автор выделяет:

1) наличие механизмов аутентификации пользователей: система должна обладать функциями аутентификации пользователей, предшествующей процессу выполнения ими пользовательских запросов;

2) наличие механизмов разграничения прав доступа к информации: система должна обеспечивать защиту он несанкционированного доступа к информации, хранящейся в ней путем использования механизмов разграничения прав на чтение/запись информации пользователями;

3) наличие механизмов построения защищенных каналов связи: в системе должна быть предусмотрена возможность использования современных протоколов защищенного обмена данными, такими как TLS 2.0/SSL 3.0;

4) обеспечение конфиденциальности информации, хранящейся в системе: необходимо реализовать набор функций по обеспечению конфиденциальности информации, хранимой в системе, путем использования криптографических средств;

5) поддержка криптографических функций: система должна содержать в себе криптографическое ядро, функции которого используются при выполнении криптографических преобразований над информацией, хранящейся в системе.

2. Общая архитектура решения

В соответствии с описанными выше требованиями необходимо разработать систему, которая бы позволяла осуществлять безопасное управление данными, хранящимися в системе и построенными по многомерному принципу. Требуется разработать систему, которая бы позволила обеспечивать конфиденциальность информации (и, в частности, защиту от НСД), учитывая при этом специфику построения многомерных структур данных. В первую очередь необходимо выделить логические компоненты, которые необходимо включить систему для того, чтобы обеспечить базовые функции последней. Выделяются следующие логические компоненты системы:

- Сервис Атрибутов (СА);
- Прокси-Сервис (ПС);
- Сервис Контроля Доступа (СКД);
- Хранилище Данных (ХД).

Наличие такого минимального набора компонентов, взаимодействующих друг с другом при помощи протоколов безопасности, позволяет поддержать запрашиваемый набор функций, обеспечивая при этом надлежащий уровень безопасности. Предлагаемая архитектура изображена на рис. 1. Далее каждый из компонентов описывается более подробно.

Рис. 1. Архитектура безопасной системы управления многомерными структурами данных (Fig. 1. Proposed secure multidimensional data management system architecture)

3. Сервис Атрибутов

Первым рассматриваемым компонентом станет Сервис Атрибутов. Его наличие в системе обуславливается необходимостью хранения наборов атрибутов, соответствующих каждой учетной записи в системе. Помимо этого на сервис возлагаются функции управления учетными данными пользователей, а также идентификации/аутентификации пользователей и управления ролями. В соответствии с предложенной парадигмой разделения прав доступа за каждым пользователем в системе закрепляется некоторый набор атрибутов, позволяющий получить пользователю доступ к тем или иным данным, хранящимся в системе. При этом необходимо обеспечивать связь набора атрибутов и учетной записи того или иного пользователя. Таким образом, как ясно из названия, основной задачей этого компонента является управление атрибутами пользователей.

В дополнение к этому сервис является источником информации о наборах атрибутов пользователей для других компонентов системы. Подобное поведение сервиса необходимо для того, чтобы другие компоненты системы, основываясь на информации об атрибутах, полученных от Сервиса, могли сделать вывод о том, следует ли предоставлять право на чтение информации пользователю.

Из всего сказанного выше можно сделать вывод об архитектуре данного компонента. Компонент должен состоять из специализированного программного обеспечения, обеспечивающего управление информацией об атрибутах, а также в его состав необходимо включить Сервер Управления базами данных, на котором будет храниться информация об учетных записях пользователей и соответствующих им наборах атрибутов. Схема базы данных, хранящей информацию о пользователях, должна включать в себя следующие таблицы:

- Пользователь – таблица, содержащая в себе базовые учетные данные пользователя, такие как идентификатор и аутентификационные данные;
- Роль – таблица, содержащая в себе описание всех ролей в системе;
- Атрибуты – таблица, каждая строка которой содержит в себе набор атрибутов, закрепленный за пользователем;
- Пользовательская роль – таблица, которая обеспечивает логическое отношение Пользователь – роль, то есть хранит в себе информацию о том, какие роли из доступных в системе были назначены тому или иному пользователю;

- Пользовательские атрибуты – таблица, связывающая между собой учетную запись пользователя и соответствующий ей набор атрибутов, хранящийся в таблице Атрибуты.

Программное обеспечение, осуществляющее управление учетными данными пользователя должно поддерживать следующий минимальный набор требуемых функций:

- Регистрация пользователя – ПО должно обладать механизмами, позволяющими создать в системе новую учетную запись;
- Удаление пользователя – ПО должно обладать функцией удаления учетной записи пользователя системы и всей связанной с ней информации (такой как принадлежность к роли и набор атрибутов, назначенный данной учетной записи);
- Управление ролями – ПО должно обладать возможностью назначения учетной записи пользователя роли системы, целиком описывающей то, какие действия пользователь может выполнять в системе;
- Аутентификация – в составе ПО Сервиса Атрибутов должны присутствовать механизмы, позволяющие проводить процедуру аутентификации пользователей, чьи учетные записи хранятся на сервисе;
- Назначение атрибутов – ПО должно обладать механизмами, позволяющими сопоставить пользователя системы и набор атрибутов, однозначно определяющий права на чтение данных в системе;
- Поиск по атрибутам пользователя – ПО должно включать в себя возможность поиска информации об атрибутах доступа, связанных с пользователем из запроса, и предоставления этой информации авторизованным третьим лицам;
- Поддержка криптографических функций – ПО должно включать в себя реализацию всех криптографических примитивов, используемых в системе.

Исходя из специфики выполняемых Сервисом Атрибутов функций, он должен находиться в рамках контролируемой зоны. Это обуславливается тем, что в рамках данного сервиса хранится информация, утечка которой может поставить под угрозу компрометации всю систему целиком. Таким образом, доступ к Сервису Атрибутов должен осуществляться только для авторизованных пользователей системы, прошедших процедуру регистрации и аутентификации.

4. Прокси-Сервис

Данный компонент является логическим ядром всей системы. К основным функциям этого компонента относится в первую очередь криптографическое преобразование элементов запроса для обеспечения его конфиденциальности, а также управление остальными компонентами системы. Прокси-Сервис должен обеспечивать прозрачную для пользователя работу с информацией, хранящейся у поставщика услуг.

При поступлении запроса от пользователя сервис осуществляет его синтаксический анализ. В результате этого процесса сервис получает информацию о том, к каким ячейкам пользователь пытается получить доступ. Данная информация используется Прокси-Сервисом в первую очередь для того, чтобы принять решение о том, имеет ли доступ право на получение запрошенной информации. Для этого Прокси-Сервис обращается к Сервису Атрибутов и Сервису Контроля Доступа с целью получения информации о наборах атрибутов пользователя и запрашиваемых им ячеек. На основании полученной информации Прокси-Сервис принимает решение о выполнении запроса. В случае положительного решения Прокси-Сервис осуществляет криптографические преобразования полей запроса в соответствии со схемой многомерного куба данных, хранящихся в системе. Данные преобразования позволяют обеспечить конфиденциальность запроса и не позволяют злоумышленнику получить информацию о схеме куба данных (измерения и иерархии). Также в случае получения ответа от поставщика услуг, содержащего зашифрованные данные из куба, Прокси-Сервис

осуществляет сбор всех необходимых ключей расшифрования и передачу их пользователю вместе с результатом выполнения запроса.

Таким образом, Прокси-Сервис представляет собой совокупность программного комплекса, реализующего основные сервисные функции, и сервера баз данных, осуществляющего хранение служебной информации. К последней относится схема многомерного куба, содержащая сведения о структуре куба, его мерах, измерениях и иерархиях. В дополнение к этому Прокси-Сервис должен хранить ключи шифрования, использованные при шифровании значений измерений и иерархий. Данная информация необходима Прокси-Сервису для обеспечения возможности преобразования запросов, поступающих от пользователей и сокрытия конфиденциальной информации, которая может храниться в таких запросах.

Программное обеспечение, используемое на Прокси-сервисе, должно обладать следующим набором функций:

- Преобразование пользовательских запросов – ПО должно включать в себя функции, позволяющие выполнить анализ и разбор пользовательского запроса, поступившего на Сервис;
- Получение информации об атрибутах пользователя – ПО должно реализовывать механизмы, позволяющие выполнять запросы к Сервису Атрибутов с целью получения информации о наборе атрибутов, присвоенных пользователю, осуществляющему запрос на чтение данных;
- Получение информации о политиках безопасности данных, входящих в запрос – в ПО должна присутствовать возможность обращения к Сервису Контроля Доступа с целью получения информации о том, каким набором атрибутов должен обладать пользователь для доступа к запрашиваемой информации;
- Контроль доступа – ПО должно обладать возможностью принятия решения о том, обладает ли пользователь достаточными правами на информацию, доступ к которой необходимо получить в рамках запроса, поступившего на Прокси-сервис. Решение принимается на основании информации, полученной от Сервиса Атрибутов и Сервиса Контроля Доступа;
- Чтение/запись информации из хранилища Поставщика Услуг – ПО должно включать в себя функции, позволяющие посредством интерфейсов, предоставляемых Поставщиком Услуг, осуществлять чтение и запись информации в хранилища данных, расположенные на стороне Поставщика Услуг;
- Подготовка ключевой информации – в список функций, реализуемых ПО Сервиса должны входить функции, позволяющие определить необходимый набор ключевой информации, которая должна быть получена с Сервиса Контроля Доступа и передана пользователю вместе с запрашиваемыми им данными для того, чтобы пользователь в случае обладания достаточными правами мог выполнить корректное расшифрование информации, полученной в результате выполнения пользовательского запроса;
- Поддержка криптографических функций – ПО должно реализовывать набор криптографических примитивов, необходимых для обеспечения конфиденциальности информации, содержащейся в запросе на доступ к хранилищу данных, путем модификации элементов запроса при помощи криптографических преобразований последних.

Прокси-сервис, хранящий чувствительную к утечке информацию в виде ключей шифрования, использованных при подготовке схемы многомерного куба в защищенном исполнении, необходимо размещать в рамках контролируемой зоны. Это обуславливается тем, что утечка информации с Прокси-сервиса может привести к тому, что злоумышленник получит информацию об исходной схеме многомерного куба, которая в некоторых случаях также может являться конфиденциальной информацией.

5. Сервис Контроля Доступа

Обеспечение защиты он несанкционированного доступа в системе обеспечивается путем использования атрибутного подхода к разделению доступа. Для того, чтобы эффективно управлять доступом к данным на основе атрибутов, необходимо хранить информацию как об атрибутах пользователя (это является задачей Сервиса Атрибутов), так и о наборах атрибутов, присвоенных тому или иному элементу данных, так называемых политиках доступа к данным.

Именно для достижения этих целей предлагается использовать Сервис Контроля Доступа. В спектр задач Сервиса входят запись, хранение и управление политиками доступа к ячейкам данных гиперкуба. В дополнение к этому Сервис осуществляет хранение и передачу ключей доступа к информации, запрашиваемой пользователями. В процессе обработки запроса на извлечение информации из хранилища Сервис Контроля Доступа снабжает Прокси-сервис информацией, на основании которой Прокси-сервис принимает решение о выдаче прав на чтение ячейки, а также, в случае положительного решения, обеспечивает Прокси-сервис всеми необходимыми ключами расшифрования, требуемыми пользователю для получения информации из запроса.

Таким образом, для каждой ячейки данных Сервис Контроля Доступа осуществляет хранение политики доступа к этой ячейке, представленной в виде набора атрибутов, а также симметричного ключа шифрования данных этой ячейки, зашифрованного при помощи схемы атрибутного шифрования.

6. Протокол взаимодействия СДХ и БД РЗК

Провайдер услуг выступает в качестве лица, оказывающего услуги по обработке и хранению информационных массивов. Во владении провайдера услуг находится хранилище или хранилища данных, которые представляет собой программный продукт, управляющий жизненным циклом данных, построенных по многомерному принципу. Для разрабатываемой системы хранилище данных является внешним подключаемым компонентом, а это значит, что взаимодействие с ним будет производиться через программный интерфейс хранилища данных, предоставляемый провайдером услуг.

Программный интерфейс, предоставляемый хранилищем, должен предоставлять следующий минимальный набор функций: создание нового гиперкуба данных, задание схемы гиперкуба, запись информации в гиперкуб, чтение информации из ячеек гиперкуба.

Так как разрабатываемая система подразумевает управление конфиденциальными данными, необходимо, чтобы эти данные хранились в изолированном гиперкубе и не пересекались с информацией в открытом доступе. Для этого необходимо, чтобы сторонний поставщик услуг хранения данных при помощи программного интерфейса своего хранилища данных позволял создание отдельных кубов данных.

После того, как в рамках конфигурации системы в хранилище данных будет создан гиперкуб, необходимо описать его структуру путем задания схемы данного куба. Для этого хранилище данных должно обладать возможностью модификации схемы куба извне. Задание схемы куба необходимо потому, что в рамках обеспечения конфиденциальности информации, хранимой в системе, исходная схема данных будет модифицирована при помощи криптографических преобразований, позволяющих скрыть от злоумышленника не только содержимое ячеек данных, но и структуру гиперкуба, их содержащего.

Наконец, запись и чтение информации из куба являются критическими функциями системы, поэтому присутствие подобных функций в программном интерфейсе хранилища данных является необходимым критерием корректной работы всей системы в целом. Необходимо убедиться в том, что хранилище данных обеспечивает защиту каналов связи, по которым происходит обращение к программному интерфейсу хранилища.

Заключение

В статье предложено решение задачи защищенной обработки конфиденциальной информации, построенной по многомерному принципу. Решение представлено в виде архитектуры системы, позволяющей обеспечивать безопасное выполнение операций чтения и записи конфиденциальной информации, хранящейся в многомерных кубах данных. Архитектура включает в себя описание компонентов системы, а также требования, предъявляемые к каждому из компонентов в части набора функций, которыми должен обладать этот компонент.

Практическая ценность предложенного решения состоит в расширении области применения систем управления многомерными структурами данных к прикладным задачам, требующим обработки конфиденциальной информации.

Перспективы продолжения исследования заключаются в решении задач контроля доступа (при одновременном обеспечении безопасности пользователей) к хранилищам данных более сложной структуры, в частности, к многомерным массивам данных.

СПИСОК ЛИТЕРАТУРЫ:

1. Российская Федерация. Распоряжения Правительства Российской федерации. Об утверждении Концепции перевода обработки и хранения государственных информационных ресурсов, не содержащих сведения, составляющие государственную тайну, в систему федеральных и региональных центров обработки данных [№ 1995-р от 07.10.2015 г.]. — М., 2015.
2. Российская Федерация. Распоряжения Правительства Российской федерации. Об утверждении программы «Цифровая экономика Российской Федерации» [№ 1632-р от 28.07.2017 г.]. — М., 2017.
3. Codd, E. Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate / E. Codd, S. Codd, C. Salley // Codd & Date, Inc —1993.
4. Gorlatykh, A. Challenges of Privacy-Preserving OLAP Techniques / A. Gorlatykh, S. Zapechnikov // Proceedings of the 2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus). —2017.
5. Popa R. A. CryptDB: Protecting Confidentiality with Encrypted Query Processing // Raluca Ada Popa, Catherine M. S. Redfield, Nickolai Zeldovich, and Hari Balakrishnan / In Proceedings of the 23rd ACM Symposium on Operating Systems Principles (SOSP). —2011. —Portugal.
6. Popa R. A. Cryptographic treatment of CryptDB's Adjustable Join / R. A. Popa N. Zeldovich // Technical Report MIT-CSAIL-TR-2012-006, Computer Science and Artificial Intelligence Laboratory —2012. — Cambridge.
7. Poddar R. Arx: A DBMS with Semantically Secure Encryption / R. Poddar, T. Boelter, R. A. Popa // In Technical Report No. UCB/EECS-2017-111 of University of California, Berkeley — 2006.
8. Camenisch J., Dubovitskaya M., Neven G. Oblivious transfer with access control. Proc. of ACM CCS 09, Chicago, Illinois, USA, November 9-13, 2009. ACM Press. Pp. 131-140.
9. Camenisch J., Dubovitskaya M., Neven G. Unlinkable priced oblivious transfer with rechargeable wallets. Proc. of Financial Cryptography'10. pp. 66-81.
10. Sahai A., Waters B. Fuzzy identity-based encryption. 15 pp. URL: http://eprint.iacr.org/2004/086 (дата обращения: 25.01.2017 г.)
11. Goyal V., Pandey O., Sahai A., Waters B. Attribute-based encryption for fine-grained access control of encrypted data. 28 pp. URL: http://eprint.iacr.org/2006/309 (дата обращения: 25.01.2017 г.)
12. Bethencourt J., Sahai A., Waters B. Ciphertext-policy attribute-based encryption. 15 pp. URL: http://www.cs.utexas.edu/~bwaters/publications/papers/cp-abe.pdf (дата обращения: 25.01.2017 г.)
13. Lewko, A. Decentralizing Attribute-Based Encryption / A. Lewko, B. Waters // In: Paterson K.G. Advances in Cryptology (EUROCRYPT). —Springer. —2011.

REFERENCES:

[1] Russian Federation. The order of the Government of the Russian Federation. About the approval of the Concept of transfer of processing and storage of the state information resources which are not containing the data which are the state secret to system of Federal and regional data processing centers [No. 1995-p of 07.10.2015]. —M., 2015. (in Russian).
[2] Russian Federation. The order of the Government of the Russian Federation. About the approval of the program "Digital economy of the Russian Federation" [No. 1632-p of 28.07.2017]. — M. , 2017.Codd, E. Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate E. Codd, S. Codd, C. Salley. Codd & Date, Inc —1993. (in Russian).

[3] Codd, E. Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate. E. Codd, S. Codd, C. Salley. Codd & Date, Inc —1993.
[4] Gorlatykh, A. Challenges of Privacy-Preserving OLAP Techniques. A. Gorlatykh, S. Zapechnikov. Proceedings of the 2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus). —2017.
[5] Popa R. A. CryptDB: Protecting Confidentiality with Encrypted Query Processing. Raluca Ada Popa, Catherine M. S. Redfield, Nickolai Zeldovich, and Hari Balakrishnan. In Proceedings of the 23rd ACM Symposium on Operating Systems Principles (SOSP). —2011. —Portugal.
[6] Popa R. A. Cryptographic treatment of CryptDB's Adjustable Join. R. A. Popa N. Zeldovich. Technical Report MIT-CSAIL-TR-2012-006, Computer Science and Artificial Intelligence Laboratory —2012. —Cambridge.
[7] Poddar R. Arx: A DBMS with Semantically Secure Encryption. R. Poddar, T. Boelter, R. A. Popa. In Technical Report No. UCB/EECS-2017-111 of University of California, Berkeley — 2006.
[8] Camenisch J., Dubovitskaya M., Neven G. Oblivious transfer with access control. Proc. of ACM CCS 09, Chicago, Illinois, USA, November 9-13, 2009. ACM Press. Pp. 131-140.
[9] Camenisch J., Dubovitskaya M., Neven G. Unlinkable priced oblivious transfer with rechargeable wallets. Proc. of Financial Cryptography'10. pp. 66-81.
[10] Sahai A., Waters B. Fuzzy identity-based encryption. 15 pp. URL: http://eprint.iacr.org/2004/086 (дата обращения: 25.01.2017 г.)
[11] Goyal V., Pandey O., Sahai A., Waters B. Attribute-based encryption for fine-grained access control of encrypted data. 28 pp. URL: http://eprint.iacr.org/2006/309 (date accessed: 25.01.2017 г.)
[12] Bethencourt J., Sahai A., Waters B. Ciphertext-policy attribute-based encryption. 15 pp. URL: http://www.cs.utexas.edu/~bwaters/publications/papers/cp-abe.pdf (date accessed: 25.01.2017 г.)
[13] Lewko, A. Decentralizing Attribute-Based Encryption. A. Lewko, B. Waters. In: Paterson K.G. Advances in Cryptology (EUROCRYPT). —Springer. —2011.

Поступила в редакцию - 04 апреля 2018 г. Окончательный вариант – 23 августа 2018 г.
Received – April 04, 2018. The final version – August 23,2018.