ОБЕСПЕЧЕНИЕ ЦЕЛОСТНОСТИ ДАННЫХ НА БЛОЧНОМ УРОВНЕ С ИСПОЛЬЗОВАНИЕМ МЕТОДА МАТРИЧНОЙ СВЯЗИ ДЛЯ ЗАЩИТЫ ДАННЫХ В ОБЛАЧНОМ ХРАНИЛИЩЕ

Ю. О. Трусова

ФГУП «РФЯЦ-ВНИИЭФ», г. Саров Нижегородской обл.

Облачные вычисления – это современная парадигма, позволяющая динамически распределять ресурсы Интернета с минимальными эксплуатационными затратами. Облачные хранилища позволяют пользователю экономически эффективно использовать информационные ресурсы, но возникает ряд проблем, касающихся безопасности. В частности угроза их целостности, доступности, конфиденциальности. В облачных вычислениях обеспечение целостности данных является одной из основных проблем, поскольку пользователь не может контролировать механизмы безопасности для защиты данных.

Целостность данных – это форма защиты данных от потери и повреждения, вызванных аппаратным, программным обеспечением или сбоем сети [1]. Ошибка данных может произойти случайно, за счет ошибок программирования, или быть результатом злонамеренного нарушения или взлома. Обеспечение целостности данных – одна из важнейших проблем, поскольку высокий уровень надежности способен гарантировать правильность, доступность, качество, надежность, безопасность, конфиденциальность, точность хранящихся данных, а так же позволяет пользователю быть уверенным, что информация не изменяется и не повреждается поставщиком услуг или другими пользователями.

Кроме высокого уровня надежности от современных методов обеспечения целостности данных требуется высокая производительность. Производительность измеряется с помощью таких параметров, как время вычисления, время шифрования и время дешифрования, количество используемой памяти и размер выходных данных. Пока невозможно провести аутсорсинг данных с использованием механизмов облачного хранилища, так как он не поддерживает локальную копию. Следовательно, криптографические меры не могут использоваться непосредственно для контроля целостности данных. Поэтому необходима третья сторона – внешний сторонний аудитор (TPA). TPA – это независимый орган, обладающий возможностями для мониторинга целостности данных, переданных сторонним клиентом, а также информирующий о повреждении или потере данных, если такие имеются [2]. Но для его работы требуется отдельная память, а также большее количество времени для проверки целостности данных; следовательно, общая производительность снижается. В настоящее время специалисты по программному обеспечению используют ряд практик для обеспечения целостности данных, который включает в себя шифрование данных, резервное копирование данных, средства контроля доступа, проверку ввода, проверку данных, обнаружение ошибок и коррекцию при передаче и хранении данных. Эффективность методов проверки нарушения данных зависит от служебных данных связи, издержек памяти, размера ключа, времени шифрования, времени дешифрования и времени вычисления. Целостность данных можно обеспечивать на двух уровнях: во-первых, чтобы предотвратить повреждение данных, во-вторых, обнаружить и исправить нарушение данных. В данной работе акцент сделан именно на обнаружение повреждений данных.

Предлагаемый к рассмотрению метод основан на методе вычисления определителя матрицы (DF) для повышения, как целостности данных, так и безопасности. Перед передачей серия данных разбивается на N-матриц, где N задается следующим образом

$$N = \frac{общее\ количество\ данных}{(d \times d)}, \qquad (1)$$

где (d×d) – количество элементов на матрицу. Определитель каждой матрицы вычисляется и добавляется в пакет вместе с данными. На этапе получения он сравнивается с определителем полученных данных, для обеспечения целостности.

Отмечается, что существует один недостаток данного метода – случай, когда DF равен нулю. Определитель равен нулю, если какая-либо из строк пропорциональна другой строке. То же самое верно и для столбцов, или, если одна из строк или столбец имеют только нулевые значения. Кроме того DF не отображает изменений, если некоторые из строк или столбцов взаимозаменяемые. Чтобы решить эту проблему используем метод вращения матрицы. Каждый элемент матрицы восстанавливается с матричным методом, с использованием исходной матрицы и преобразованной матрицы. Наборы вращательных матриц вычисляются и добавляются к каждой исходной матрице данных.

Например, значение DF для следующей матрицы равно нулю. Применяя предложенный метод, мы получаем новую матрицу, определитель которой уже не является нулем (см. рис.1).

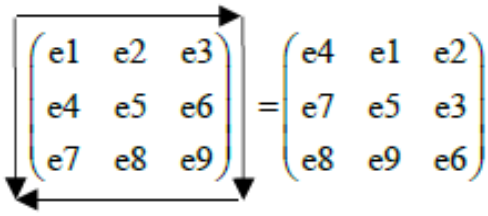

Рис. 1. Применение метода вращения к матрице (3×3)

Затем для каждого полученного определителя с помощью комбинации алгоритмов SHA-1 и AES формируем цифровую подпись. В конце перед самой передачей или хранением данных в облаке для повышения безопасности применяем раскрашивание данных. На стороне приемника оба детерминанта снова пересчитываются, и дегенерируют хэш-последовательность, затем сравниваются со значениями отправителя. Совпадение данных гарантирует, что не было изменений во время передачи, иначе были бы нарушены конкретные блоки принятых данных. Результаты предлагаемой системы показывают, что метод матричной связи превосходит другие метод проверки целостности данных, а также обеспечивает конфиденциальность данных, их защиту от несанкционированных пользователей. Рис. 2 показывает архитектуру предлагаемой системы.

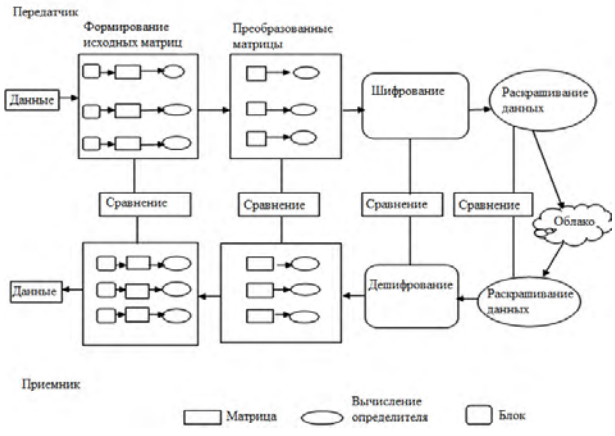

Рис. 2. Алгоритм матричного метода обеспечения целостности данных

Приведем основные этапы предлагаемого метода:

– передатчик:

1) исходные данные поступают в виде строки. Каждая строка преобразуется в байтовый формат, разделение на блоки;

2) преобразование каждого блока в квадратную матрицу;

3) вычисление определителя для каждой матрицы;

4) постройка новой матрицы с использованием метода матричного набора на основе блоков, чтобы исключить нулевой DF;

5) вычисление определителя для матриц, построенных на этапе 4;

6) генерация хэш-функции алгоритмом SHA-1 для всех вычисленных определителей;

7) шифрование значений хэширования с помощью алгоритма AES для генерации цифровой подписи;

8) применение раскрашивания данных для каждой цифровой подписи, полученной на этапе 7;

9) хранение цветных данных в облачном хранилище.

– приемник:

1) восстановление цвета из цветных данных;

2) декодирование полученных данных;

3) восстановление преобразованных матриц;

4) вычисление их определителей;

5) восстановление исходных матриц и вычисление их определителей;

6) сравнение результатов, полученных на этапах 1, 2, 4, 5 отправителя с этапами 8, 6, 5, 3 приемника соответственно;

7) если результаты одинаковые на всех сравниваемых шагах, то целостность данных не нарушена. Если на одном из шагов данные различаются, то нарушен конкретный блок данных.

Неоднократное сравнение результатов, полученных на разных этапах вычисления, обеспечивает большую надежность, так как позволяет минимизировать вероятность ошибки.

Для подтверждения теоретической модели проведем моделирование алгоритма матричного метода для данных разного размера. Результаты экспериментов можно представить в виде таблицы, показывающей точности с точки зрения количества дефектов, обнаруженных для разных размеров данных.

Точности обнаружения дефектов для блоков данных, разного размера

Размер данных в байтах	Фактическое количество дефектных блоков	Количество дефектных блоков, обнаруженных предложенным методом	Точность предлагаемого метода (%)
10000	08	08	100
15000	10	09	99,91
20000	12	12	100
22000	14	13	99,91
30000	17	15	99,66
33000	19	19	100

Графически полученные результат представлены на рис. 3

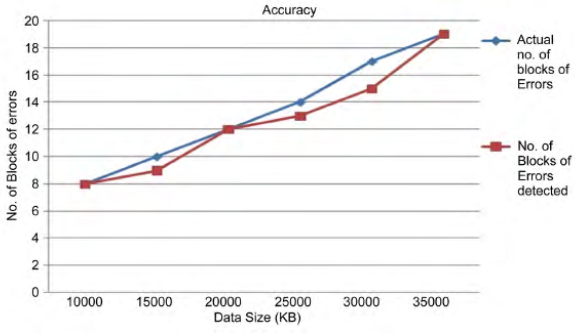

Рис. 3. Проверка точности обнаружения дефектов для блоков данных, разного размера

В докладе представлен способ повышения безопасности данных путем улучшения проверки целостности данных в облачном хранилище без использования TPA. Исходные данные делятся на блоки, где каждый блок помещается в квадратную матрицу. Элементы матрицы помещаются в новую форму с использованием хэширования, которое приводит к экономии памяти за счет уменьшения битов, а также для повышения точности данных.

Кроме того, цифровая подпись применяется к каждому определителю матрицы для повышения надежности данных. Эта модель также использует раскраску данных для шифрованной цифровой подписи для повышения безопасности данных, которая помогает пользователю проверять наличие несанкционированного доступа к данным.

В предлагаемом методе точность поддерживается на высоком уровне путем двухкратной перестановки данных. Перестановка осуществляется через исходную и преобразованную матрицы. Хотя для этого требуется более долгие вычисления, зато обеспечивается высокий уровень точности и безопасности данных. Таким образом, здесь представлена попытка обеспечить новое понимание безопасности облачного хранилища путем обнаружения нарушений целостности данных на уровне блоков во время хранения или передачи.

Литература

1. Kahate, A. Cryptography and Network Security. New Delhi:Tata McGraw-Hill Publishing Company, 2008.

2. Govinda, K., Gurunathprasad, V. and Sathishkumar, H. Third Party Auditing for Secure Data Storage in Cloud through Digital Signature Using RSA. // International Journal of Advanced Scientific and Technical Research,2012,Vol 4

3. Camara, L., Li, J., Li, R. and Kagorora, F. Block-Based Scheme for Database Integrity Verification. // International Journal of Security and Its Applications, 2014,Vol.8, P.25 – 40.

4. Дроздова И. И., Жилин В. В. Безопасность облачных хранилищ // Технические науки в России и за рубежом: материалы VIIМеждунар. науч. конф. – М.: Буки-Веди, 2017. – С. 16–18.

5. Stallings, W. Cryptography and Network Security. 4th Edition, Pearson Prentice Hall, 2006.