(19) Евразийское патентное ведомство

(11) 040281 (13) B1

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ЕВРАЗИЙСКОМУ ПАТЕНТУ

(45) Дата публикации и выдачи патента
2022.05.17

(21) Номер заявки
202092868

(22) Дата подачи заявки
2020.12.23

(51) Int. Cl. ***G06F 17/00*** (2019.01)
G06N 20/00 (2019.01)
G06F 21/45 (2013.01)

(54) СПОСОБ ТОКЕНИЗАЦИИ НОМЕРА БАНКОВСКОЙ КАРТЫ И СПОСОБ ДЕТОКЕНИЗАЦИИ НОМЕРА БАНКОВСКОЙ КАРТЫ

(31) **2020131500**
(32) **2020.09.24**
(33) **RU**
(43) **2022.03.31**

(56) EP-B1-2927836
US-B2-8739262
US-B2-10560451
US-A1-2020213121

(71)(73) Заявитель и патентовладелец:
ПУБЛИЧНОЕ АКЦИОНЕРНОЕ ОБЩЕСТВО "СБЕРБАНК РОССИИ" (ПАО СБЕРБАНК) (RU)

(72) Изобретатель:
Коршунов Ян Юрьевич, Соколов Дмитрий Юрьевич, Сафронов Юрий Викторович, Никитин Александр Юрьевич (RU)

(74) Представитель:
Герасин Б.В. (RU)

(57) Заявленное техническое решение относится к области кодирования и декодирования данных, а в частности к автоматизированным способам и системам токенизации номера банковских карт и определения детокенизированного номера банковских карт. Техническим результатом, достигающимся при решении данной проблемы, является повышение эффективности защиты номеров банковских карт за счет нового способа токенизации. Указанный технический результат достигается благодаря осуществлению компьютерно-реализуемого способа токенизации номера банковской карты (PAN), выполняемый с помощью по меньшей мере одного процессора и содержащий этапы, на которых получают PAN и осуществляют его разбиение на составляющие части: бин (BIN), состоящий из первых шести символов, маска (MASK), состоящая из четырех последних символов, и средняя часть (mPAN); определяют значение сдвига в таблице замен, при котором вычисляют первое значение сдвига по таблице замен с помощью криптопреобразования конкатенированной строки значений BIN и MASK; вычисляют второе значение сдвига по таблице замен с помощью криптопреобразования mPAN; вычисляют суммарное значение сдвига путем сложения первого и второго значений сдвига; вычисляют индекс в таблице замен с помощью нормализации суммарного значения сдвига по таблице замен, причем нормализация выполняется по размерности таблицы как остаток от деления смещения на количество строк таблицы, получая при этом искомый индекс в таблице замен; с помощью полученного индекса выбирают из таблицы замен токенизированную часть номера zPAN для замены mPAN; формируют токинезированный номер карты DPAN с помощью замены mPAN на zPAN, используя BIN и MASK полученного номера PAN.

040281 B1

040281 B1

Область техники

Заявленное техническое решение, в общем, относится к области кодирования и декодирования данных, а в частности к автоматизированным способам и системам токенизации номера банковских карт и детокенизации номера банковских карт.

Уровень техники

Одной из самых важных задач в информационной безопасности является защита конфиденциальных данных, которая является комплексной и довольно сложной. Организация эффективной защиты конфиденциальных данных в банковских системах, платежных приложениях, базах данных, устройств хранения и групп серверов сама по себе является непростой и еще больше усложняется при использовании разнородных систем.

Как правило, в случае с конфиденциальными данными рассматриваются разнообразные решения с использованием шифрования, но в последние годы заметен рост интереса к другой технологии - токенизации.

Должным образом реализованное шифрование является одним из наиболее эффективных средств обеспечения безопасности конфиденциальных данных. Оно позволяет обеспечить доступ к данным только авторизованным пользователям и защищает данные как при хранении, так и при передаче. Но шифрование не единственный вариант защиты - есть и альтернативные методы. Причем иногда самым правильным решением будет не попытка защитить конфиденциальные данные шифрованием, а вообще отказаться от их передачи.

Токенизация как раз является технологией, которая предоставляет эту возможность - ее принцип заключается в подмене реальных конфиденциальных данных некими значениями - токенами. Токенизация в чем-то схожа с шифрованием - обе технологии занимаются маскированием реальных данных, но подход к этому процессу в корне отличается. В случае шифрования процесс сокрытия данных обратим при наличии правильного ключа. Любой человек, получив ключ шифрования, сможет восстановить исходные данные.

В случае с токенизацией процесс не является обратимым, ведь вместо данных при токенизации используется токен, не несущий в себе никакой конфиденциальной информации, или результатом ее криптопреобразования и лишь логически связанный с реальными данными, которые хранятся в хорошо защищенной базе данных. При этом токен может иметь тот же формат (размер и структура), что и оригинальные данные, что позволяет минимизировать необходимость внесения изменений в работающие с ним приложения. Но при этом похищать токен бессмысленно, так как он не поможет получить никакие реальные конфиденциальные данные без доступа к системе токенизации.

Главное преимущество токенизации в том, что токены могут полноценно использоваться внутри среды их применения, но совершенно бесполезны вне ее. Токенизация идеально подходит для защиты конфиденциальных данных, таких как номера банковских карт, номера социального страхования, а также любых других данных, которые злоумышленники регулярно похищают с целью использования или продажи на "черном рынке". Если злоумышленникам не удастся взломать сам сервер токенизации, чтобы получить связанные с токенами реальные данные, то похищенные токены не дадут им ровным счетом никаких возможностей их использовать.

Повышение интереса к токенизации связано, прежде всего, с тем, что она позволяет обеспечить защиту данных при низкой нагрузке на системы организации. Добавление шифрования в системы - особенно в уже действующие - значительно увеличивает нагрузку на них. Внесение изменений в приложения для внедрения шифрования снижает производительность и увеличивает требования к разработчикам и администраторам систем. Кроме того, в случае использования разнородных систем, возникает необходимость шифрования, расшифрования и повторного шифрования данных в различных местах, что создает дополнительные уязвимости, которые могут быть использованы злоумышленниками. Причем чем больше ключей используется в системе, тем больше возможностей для ее атаки. Работа с ключами особенно опасна, учитывая распространенность вредоносного программного обеспечения (далее - ПО) для перехвата данных непосредственно в оперативной памяти, которое позволяет получить доступ к ключам, даже не имея административных привилегий на зараженной машине.

Помимо минимизации требований к внесению изменений в используемые приложения токенизация снижает риски для конфиденциальных данных. При правильной реализации приложения смогут использовать токены во всей системе и обращаться к защищенным конфиденциальным данным только в случае крайней необходимости.

Приложения могут хранить, использовать и совершать транзакции, оперируя только токеном и не подвергая реальные данные риску.

Например, одним из самых распространенных примеров использования токенизации является ее применение для платежей с использованием платежных карт. Использование токена вместо значения номера платежной карты позволяет провести отслеживание транзакции и ее выполнение без риска компрометации реальных данных карты. Доступ к реальному номеру понадобится только в контуре процессинга и при взаимодействии с платежными системами и другими участниками платежных систем. Ну а в том случае, если процессинговый центр также использует токенизацию, то возможно проведение внут-

ренней транзакции вообще без использования реальных данных.

Из уровня техники известен патент US 10262128 B2 "Tokenized data security", патентообладатель Sabre GLBL Inc, опубликовано 06.02.2014. В данном решении описывается способ токенизации номера учетной записи пользователя для защиты от кибератак с использованием карты токенов.

Недостатком известного решения в данной области техники является недостаточная защищенность данных.

Раскрытие изобретения

В заявленном техническом решении предлагается новый подход в решении технической проблемы, заключающейся в более защищенном методе хранения данных банковских карт.

Техническим результатом, достигающимся при решении данной проблемы, является повышение эффективности защиты номеров банковских карт.

Указанный технический результат достигается благодаря осуществлению компьютерно-реализуемого способа токенизации номера банковской карты (PAN), выполняемый с помощью по меньшей мере одного процессора и содержащий этапы, на которых

получают PAN и осуществляют его разбиение на составляющие части: бин (BIN), состоящий из первых шести символов, маска (MASK), состоящая из четырех последних символов, и средняя часть (mPAN);

определяют значение сдвига в таблице замен, при котором

вычисляют первое значение сдвига по таблице замен с помощью криптопреобразования конкатенированной строки значений BIN и MASK;

вычисляют второе значение сдвига по таблице замен с помощью криптопреобразования mPAN;

вычисляют суммарное значение сдвига путем сложения первого и второго значений сдвига;

вычисляют индекс в таблице замен с помощью нормализации суммарного значения сдвига по таблице замен, причем нормализация выполняется по размерности таблицы как остаток от деления смещения на количество строк таблицы, получая при этом искомый индекс в таблице замен;

с помощью полученного индекса выбирают из таблицы замен токенизированную часть номера zPAN для замены mPAN;

формируют токенизированный номер карты DPAN с помощью замены mPAN на zPAN, используя BIN и MASK полученного номера PAN.

В одном из частных вариантов реализации способа алгоритм криптопреобразования представляет собой алгоритм FF3.

Заявленное техническое решение осуществляется также за счет выполнения компьютерно-реализуемого способа детокенизации номера банковской карты (DPAN), полученного с помощью вышеуказанного способа, выполняемого с помощью по меньшей мере одного процессора и содержащего этапы, на которых

получают DPAN и осуществляют его разбиение на составляющие части: бин (BIN), состоящий из первых шести символов, маска (MASK), состоящая из четырех последних символов, и средняя часть (zPAN);

определяют значение сдвига в таблице замен, при котором

вычисляют первое значение сдвига по таблице замен с помощью криптопреобразования строки значений BIN и MASK;

вычисляют модуль первого значения сдвига по таблице замен путем разделения первого значения сдвига на размер таблицы сдвига;

вычисляют индекс в таблице замен путем поиска строки по полученному значению zPAN;

вычисляют второе значения смещения в таблице замен на основе рассчитанного индекса, модуля первого значения и размерности таблицы;

вычисляют mPAN с помощью обратного криптопреобразования от полученного второго значения смещения с использованием ключа шифрования, с помощью которого выполнялось преобразование PAN в DPAN;

формируют значение PAN на основании полученного mPAN, а также значений BIN и MASK.

В одно из частных вариантов реализации способа алгоритм криптопреобразования представляет собой алгоритм FF3.

В другом частном варианте реализации способа, если второе значение смещения отрицательное, то итеративно повторяют шаг его вычисления, инкрементировав модуль первого значения смещения на единицу.

Заявленное техническое решение также осуществляется с помощью компьютерной системы, которая содержит процессор и память, в которой хранятся машиночитаемые инструкции, выполняемые процессором для осуществления вышеуказанных способов.

Краткое описание чертежей

Признаки и преимущества настоящего технического решения станут очевидными из приводимого ниже подробного описания и прилагаемых чертежей.

Фиг. 1 иллюстрирует блок-схему способа токенизации номера банковской карты (PAN).

Фиг. 2 иллюстрирует блок-схему способа детокенизации номера банковской карты (DPAN).

Фиг. 3 иллюстрирует пример общего вида вычислительной системы, которая обеспечивает реализацию заявленного решения.

Осуществление изобретения

Ниже будут описаны понятия и термины, необходимые для понимания данного технического решения.

Токенизация - процесс замены конфиденциального элемента данных на неконфиденциальный эквивалент, называемый токеном, который не имеет самостоятельного смысла/значения для внешнего или внутреннего использования.

Токен - ссылка (т.е. идентификатор), которая сопоставляется с конфиденциальными данными через систему токенизации.

Детокенизация - обратный процесс получения значения PAN-кода по целевому токену.

PCI DSS (Payment Card Industry Digital Security Standard) - стандарт безопасности платежных карт.

PAN (Primary Account Number) - номер платежной карты.

DPAN (Digitized PAN) - токенизированный номер платежной карты.

BIN (Bank Identification Number) - идентификационный номер банка выпустившего карту.

MASK (Маска) - последние 4 символа карты, свободно использующиеся для печати на чеках и, совместно с BIN, идентификации карты в пределах одного клиента.

ЦОД - центр обработки данных.

PCI SSC (Payment Card Industry Security Standards Consul) - совет по стандартам безопасности платежных карт.

HSM (Hardware Security Module) - аппаратное решение для генерации ключей шифрования и выполнения криптопреобразований.

Алгоритм Луна (Luna Algorithm) - алгоритм определения валидности номера карты на основе десятичной целостности.

Данное техническое решение может быть реализовано на компьютере, в виде автоматизированной информационной системы (АИС) или машиночитаемого носителя, содержащего инструкции для выполнения вышеупомянутого способа.

Техническое решение может быть реализовано в виде распределенной компьютерной системы.

В данном решении под системой подразумевается компьютерная система, ЭВМ (электронно-вычислительная машина), ЧПУ (числовое программное управление), ПЛК (программируемый логический контроллер), компьютеризированные системы управления и любые другие устройства, способные выполнять заданную, чётко определённую последовательность вычислительных операций (действий, инструкций).

Под устройством обработки команд подразумевается электронный блок либо интегральная схема (микропроцессор), исполняющая машинные инструкции (программы).

Устройство обработки команд считывает и выполняет машинные инструкции (программы) с одного или более устройства хранения данных, например таких устройств, как оперативно запоминающие устройства (ОЗУ) и/или постоянные запоминающие устройства (ПЗУ). В качестве ПЗУ могут выступать, но не ограничиваясь, жесткие диски (HDD), флеш-память, твердотельные накопители (SSD), оптические носители данных (CD, DVD, BD, MD и т.п.) и др.

Программа - последовательность инструкций, предназначенных для исполнения устройством управления вычислительной машины или устройством обработки команд.

Посчитаем объем хранимых данных для всех карт, в теории существующих в индустрии: номера карт бывают от 13 до 19 символов и содержат практически все комбинации. Не будем учитывать алгоритм Луна и получим цифру около 10000 Петабайт при условии 1 байт на номер карты

$$V = \sum_{i=13}^{19} 10^i = 11\,111\,110\,000\,000\,000\,000 \text{ или около } 10\,000 \text{ Пб}$$

Получаем недостижимый объем данных и хранить такое количество информации более чем избыточно. Даже если сократить неиспользуемые BIN, объем хранения не станет приемлемым.

Выходом может стать хранение не всех элементов карты, а только заменяемых фрагментов (кроме первых 6 и последних 4 символов). В этом случае формула будет другой

$$V = \sum_{i=13}^{19} 10^{i-10} = 1\,111\,111\,000 \text{ или около } 1 \text{ Тб}$$

Объем 1 Терабайт нужно еще умножить на длину хранимой строки в базе, т.е. длины самого PAN и необходимого индекса, примерно 30 байт. Итого получается 30 Тб.

Данный объем данных тоже является большим, но с таким объёмом можно работать и провести его оптимизацию.

При замене одинаковых заменяемых фрагментов разных карт результаты замененных фрагментов также будут одинаковыми, что приведет к уязвимости реализации и постепенной накапливающейся ком-

прометации карточных данных. И единичные случаи известности связки PAN-DPAN будут автоматически компрометировать массово тысячи других номеров карт.

Рассмотрим структуру номера карты. Номер карты (PAN) состоит из BIN (первые 6 символов), MASK (последние 4 символа) и оставшейся части, подлежащей токенизации. Назовем исходное значение этой части mPAN, токенизированное - zPAN.

Для того чтобы для разных комбинаций BIN·MASK (здесь и далее символ "·" обозначает конкатенацию строк, а символ "+" арифметическую сумму) при одинаковых mPAN были разные zPAN, необходимо выборку из таблицы замены делать по некому алгоритму сдвига, гарантирующему отсутствие коллизий.

Рассмотрим таблицу замен на примере популярной длины карты =16. Для выполнения вышеописанных требований к реализации по выбранной стратегии [3], таблица должна обеспечить замену 10^6 фрагментов PAN. Так как в заменяемых символах необходимо использовать шестнадцатеричные символы, но исключить комбинации, не содержащие шестнадцатеричные цифры (A-F), то длина таблицы замен будет 16^6-10^6, т.е. 15,8 млн значений токенов для 10 млн исходных данных.

Задача состоит в определении индекса в таблице замены. Простой индекс, соответствующий порядку следования mPAN, позволит делать подбор номеров карт и токенов с таким же mPAN при других значениях BIN·MASK, что недопустимо.

Индекс со статическим сдвигом также позволит делать подбор при достаточной выборке соответствий PAN и DPAN (или mPAN и zPAN).

Следовательно, индекс должен быть динамический, т.е. не имеющий понятной зависимости от последовательности BIN·MASK, но не создающий коллизии, т.е. исключающий повторный выбор zPAN при разных mPAN, иначе говоря, не должен быть основан на случайной величине.

Для обеспечения уникальности и предотвращения коллизии необходимо осуществить ряд действий.

Сделать составной динамический индекс. Первая часть будет формироваться из неизменного значения BIN·MASK для всего диапазона mPAN. Вторая часть формируется из самой заменяемой части mPAN.

Формирование собственно индекса или сдвига (назовем его SHIFT) возможно непосредственно из этих компонентов по формуле (BIN·MASK)+mPAN. Полученное число SHIFT, нормированное по длине таблицы замен SHIFT=((BIN·MASK)+mPAN) % length({zPAN}) уже обеспечит уникальность, но пока что имеет понятную зависимость, т.е. при увеличении MASK на 1, SHIFT также увеличится на 1, что может привести к подбору связок PAN и DPAN.

Непонятной зависимости можно достичь криптопреобразованием с сохранением формата исходных данных (Format Preserving Encryption) [4], когда на выходе функции получается другой псевдослучайный элемент из конечного массива исходных данных.

Используем реализацию алгоритма FF3 [5] для выбора псевдослучайного индекса из набора индексов с использованием секретной величины и начального значения индекса. Использование в качестве базы сдвига FF3(BIN·MASK) позволит делать как прямое, так и обратное преобразование PAN-DPAN и DPAN-PAN, так как один из компонентов сдвига известен всегда (BIN·MASK), он неизменен в PAN и DPAN.

Как показано на фиг. 1, заявленный способ токенизации номера банковской карты (PAN) (100) состоит из нескольких этапов, выполняемых по меньшей мере одним процессором.

На этапе (101) получают PAN и осуществляют его разбиение на составляющие части: бин (BIN), состоящий из первых шести символов, маска (MASK), состоящая из четырех последних символов, и средняя часть (mPAN).

Далее на этапе (102) определяют значение сдвига в таблице замен. В ходе определения значения сдвига в таблице замен

на этапе (103) вычисляют первое значение сдвига по таблице замен с помощью криптопреобразования конкатенированной строки значений BIN и MASK;

на этапе (104) вычисляют второе значение сдвига по таблице замен с помощью криптопреобразования mPAN;

на этапе (105) вычисляют суммарное значение сдвига путем сложения первого и второго значений сдвига.

Далее на этапе (106) вычисляют индекс в таблице замен с помощью нормализации суммарного значения сдвига по таблице замен, причем нормализация выполняется по размерности таблицы как остаток от деления смещения на количество строк таблицы, получая при этом искомый индекс в таблице замен.

Далее на этапе (107) с помощью полученного индекса выбирают из таблицы замен токенизированную часть номера zPAN для замены mPAN.

Далее на этапе (108) формируют токинезированный номер карты DPAN с помощью замены mPAN на zPAN, используя BIN и MASK полученного номера PAN.

Алгоритм криптопреобразования, используемый в способе (100), представляет собой алгоритм FF3.

Как показано на фиг. 2, способ детокенизации номера банковской карты (DPAN) (200), полученного

с помощью вышеуказанного способа, процессор выполняет ряд последовательных этапов.

На этапе (201) получают DPAN и осуществляют его разбиение на составляющие части: бин (BIN), состоящий из первых шести символов, маска (MASK), состоящая из четырех последних символов, и средняя часть (zPAN).

Далее на этапе (202) определяют значение сдвига в таблице замен. В ходе определения значения сдвига в таблице замен

на этапе (203) вычисляют первое значение сдвига по таблице замен с помощью криптопреобразования строки значений BIN и MASK;

на этапе (204) вычисляют модуль первого значения сдвига по таблице замен путем разделения первого значения сдвига на размер таблицы сдвига;

на этапе (205) вычисляют индекс в таблице замен путем поиска строки по полученному значению zPAN;

на этапе (206) вычисляют второе значение смещения в таблице замен на основе рассчитанного индекса, модуля первого значения и размерности таблицы.

Далее на этапе (207) вычисляют mPAN с помощью обратного криптопреобразования от полученного второго значения смещения с использованием ключа шифрования, с помощью которого выполнялось преобразование PAN в DPAN.

Далее на этапе (208) формируют значение PAN на основании полученного mPAN, а также значений BIN и MASK.

Алгоритм криптопреобразования, используемый в способе (200), представляет собой алгоритм FF3.

Если на этапе (206) второе значение смещения отрицательное, то итеративно повторяют шаг его вычисления, инкрементировав модуль первого значения смещения на единицу.

Заявленное техническое решение обеспечивает новый способ токенизации банковских карт. При реализации заявленного технического решения достигается компактность хранения таблиц замен. Данные связки PAN-DPAN не хранятся на устройствах, носителях информации и т.д., тем самым обеспечивая невозможность компрометации. Для данного технического решения не требуется синхронная репликация между ЦОДами и экземплярами токенизатора в режиме реального времени. При неизвестном ключе шифрования, зная алгоритм и примеры токенизированных пар PAN-DPAN, становится невозможно вычислить другие связки PAN-DPAN. Использование двухкомпонентного сдвига по таблице замен в заявленном техническом решении позволяет быстро производить преобразование PAN-DPAN и DPAN-PAN.

На фиг. 3 представлен пример общего вида вычислительной системы (300), которая обеспечивает реализацию заявленных способов (100) (200) и/или является частью компьютерной системы, например сервером, персональным компьютером, частью вычислительного кластера, обрабатывающим необходимые данные для осуществления заявленного технического решения.

В общем случае, система (300) содержит объединенные общей шиной информационного обмена один или несколько процессоров (301), средства памяти, такие как ОЗУ (302) и ПЗУ (303), интерфейсы ввода/вывода (304), устройства ввода/вывода (1105) и устройство для сетевого взаимодействия (306).

Процессор (301) (или несколько процессоров, многоядерный процессор и т.п.) может выбираться из ассортимента устройств, широко применяемых в настоящее время, например таких производителей, как Intel™, AMD™, Apple™, Samsung Exynos™, MediaTEK™, Qualcomm Snapdragon™ и т.п. Под процессором или одним из используемых процессоров в системе (300) также необходимо учитывать графический процессор, например GPU NVIDIA или Graphcore, тип которых также является пригодным для полного или частичного выполнения способа, а также может применяться для обучения и применения моделей машинного обучения в различных информационных системах.

ОЗУ (302) представляет собой оперативную память и предназначено для хранения исполняемых процессором (301) машиночитаемых инструкций для выполнения необходимых операций по логической обработке данных. ОЗУ (302), как правило, содержит исполняемые инструкции операционной системы и соответствующих программных компонент (приложения, программные модули и т.п.). При этом в качестве ОЗУ (302) может выступать доступный объем памяти графической карты или графического процессора.

ПЗУ (303) представляет собой одно или более устройств постоянного хранения данных, например жесткий диск (HDD), твердотельный накопитель данных (SSD), флэш-память (EEPROM, NAND и т.п.), оптические носители информации (CD-R/RW, DVD-R/RW, BlueRay Disc, MD) и др.

Для организации работы компонентов системы (300) и организации работы внешних подключаемых устройств применяются различные виды интерфейсов В/В (304). Выбор соответствующих интерфейсов зависит от конкретного исполнения вычислительного устройства, которые могут представлять собой, не ограничиваясь, PCI, AGP, PS/2, IrDa, FireWire, LPT, COM, SATA, IDE, Lightning, USB (2.0, 3.0, 3.1, micro, mini, type C), TRS/Audio jack (2.5, 3.5, 6.35), HDMI, DVI, VGA, Display Port, RJ45, RS232 и т.п.

Для обеспечения взаимодействия пользователя с вычислительной системой (300) применяются различные средства (305) В/В информации, например клавиатура, дисплей (монитор), сенсорный дисплей, тач-пад, джойстик, манипулятор мышь, световое перо, стилус, сенсорная панель, трекбол, динамики,

микрофон, средства дополненной реальности, оптические сенсоры, планшет, световые индикаторы, проектор, камера, средства биометрической идентификации (сканер сетчатки глаза, сканер отпечатков пальцев, модуль распознавания голоса) и т.п.

Средство сетевого взаимодействия (306) обеспечивает передачу данных посредством внутренней или внешней вычислительной сети, например Интранет, Интернет, ЛВС и т.п. В качестве одного или более средств (306) может использоваться, но не ограничиваться, Ethernet карта, GSM модем, GPRS модем, LTE модем, 5G модем, модуль спутниковой связи, NFC модуль, Bluetooth и/или BLE модуль, Wi-Fi модуль и др.

Представленные материалы заявки раскрывают предпочтительные примеры реализации технического решения и не должны трактоваться как ограничивающие иные, частные примеры его воплощения, не выходящие за пределы испрашиваемой правовой охраны, которые являются очевидными для специалистов соответствующей области техники.

Источники информации.

1. Стандарт PCI DSS: https://www.pcisecuritystandards.org/documents/PCI_DSS_v3-2-1.pdf

2. Требования к токенизации данных:
https://www.pcisecuritystandards.org/documents/Tokenization_Guidelines_Info_Supplement.pdf

3. Статья в журнале Банковское обозрение о подходах к токенизации карточных данных в Сбербанке https://bosfera.ru/bo/pci-dss-ne-vrag

4. Format Preserving Encryption https://en.wikipedia.org/wiki/Format-preserving_encryption

5. Алгоритм FF3 https://eprint.iacr.org/2017/521.pdf

ФОРМУЛА ИЗОБРЕТЕНИЯ

1. Компьютерно-реализуемый способ токенизации номера банковской карты (PAN), выполняемый с помощью по меньшей мере одного процессора и содержащий этапы, на которых

получают PAN и осуществляют его разбиение на составляющие части: бин (BIN), состоящий из первых шести символов, маска (MASK), состоящая из четырех последних символов, и средняя часть (mPAN);

определяют значение сдвига в таблице замен, при котором

вычисляют первое значение сдвига по таблице замен с помощью криптопреобразования конкатенированной строки значений BIN и MASK;

вычисляют второе значение сдвига по таблице замен с помощью криптопреобразования mPAN;

вычисляют суммарное значение сдвига путем сложения первого и второго значений сдвига;

вычисляют индекс в таблице замен с помощью нормализации суммарного значения сдвига по таблице замен, причем нормализация выполняется по размерности таблицы как остаток от деления смещения на количество строк таблицы, получая при этом искомый индекс в таблице замен;

с помощью полученного индекса выбирают из таблицы замен токенизированную часть номера zPAN для замены mPAN;

формируют токенизированный номер карты DPAN с помощью замены mPAN на zPAN, используя BIN и MASK полученного номера PAN.

2. Способ по п.1, в котором алгоритм криптопреобразования представляет собой алгоритм FF3.

3. Компьютерно-реализуемый способ детокенизации номера банковской карты (DPAN), полученного с помощью способа по любому из пп.1, 2, выполняемый с помощью по меньшей мере одного процессора и содержащий этапы, на которых

получают DPAN и осуществляют его разбиение на составляющие части: бин (BIN), состоящий из первых шести символов, маска (MASK), состоящая из четырех последних символов, и средняя часть (zPAN);

определяют значение сдвига в таблице замен, при котором

вычисляют первое значение сдвига по таблице замен с помощью криптопреобразования строки значений BIN и MASK;

вычисляют модуль первого значения сдвига по таблице замен путем разделения первого значения сдвига на размер таблицы сдвига;

вычисляют индекс в таблице замен путем поиска строки по полученному значению zPAN;

вычисляют второе значение смещения в таблице замен на основе рассчитанного индекса, модуля первого значения и размерности таблицы;

вычисляют mPAN с помощью обратного криптопреобразования от полученного второго значения смещения с использованием ключа шифрования, с помощью которого выполнялось преобразование PAN в DPAN;

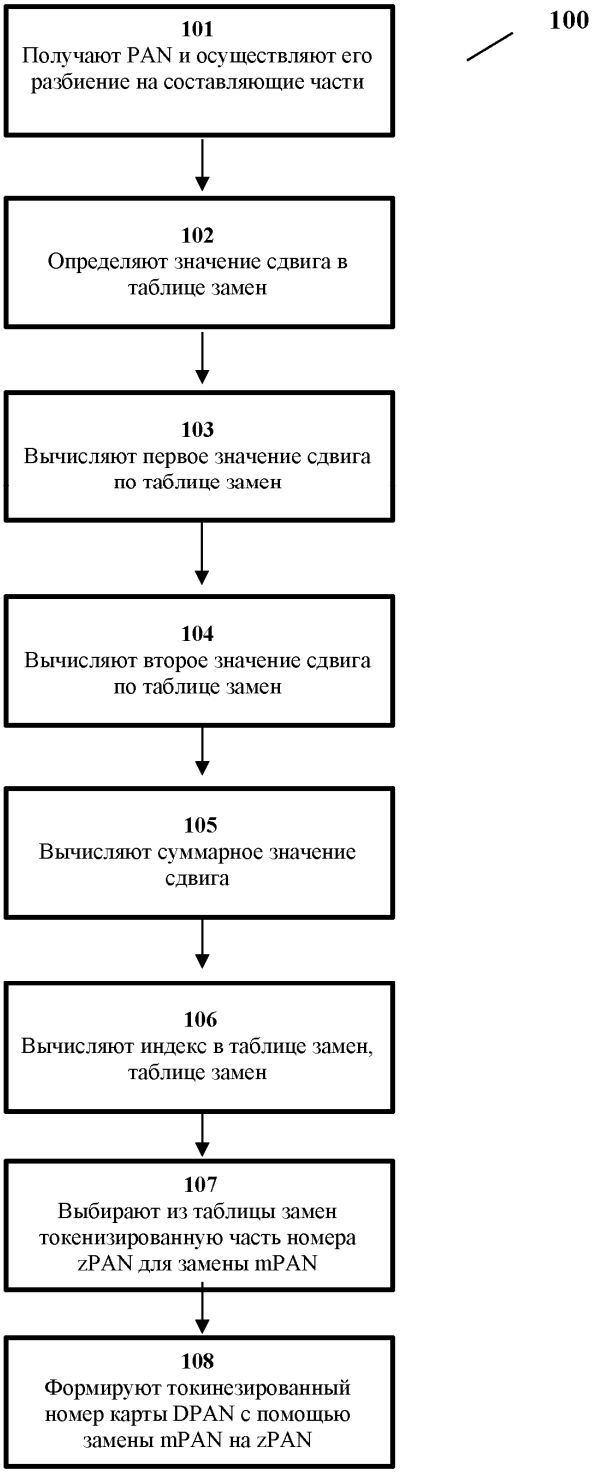
формируют значение PAN на основании полученного mPAN, а также значений BIN и MASK.

4. Способ по п.3, в котором алгоритм криптопреобразования представляет собой алгоритм FF3.

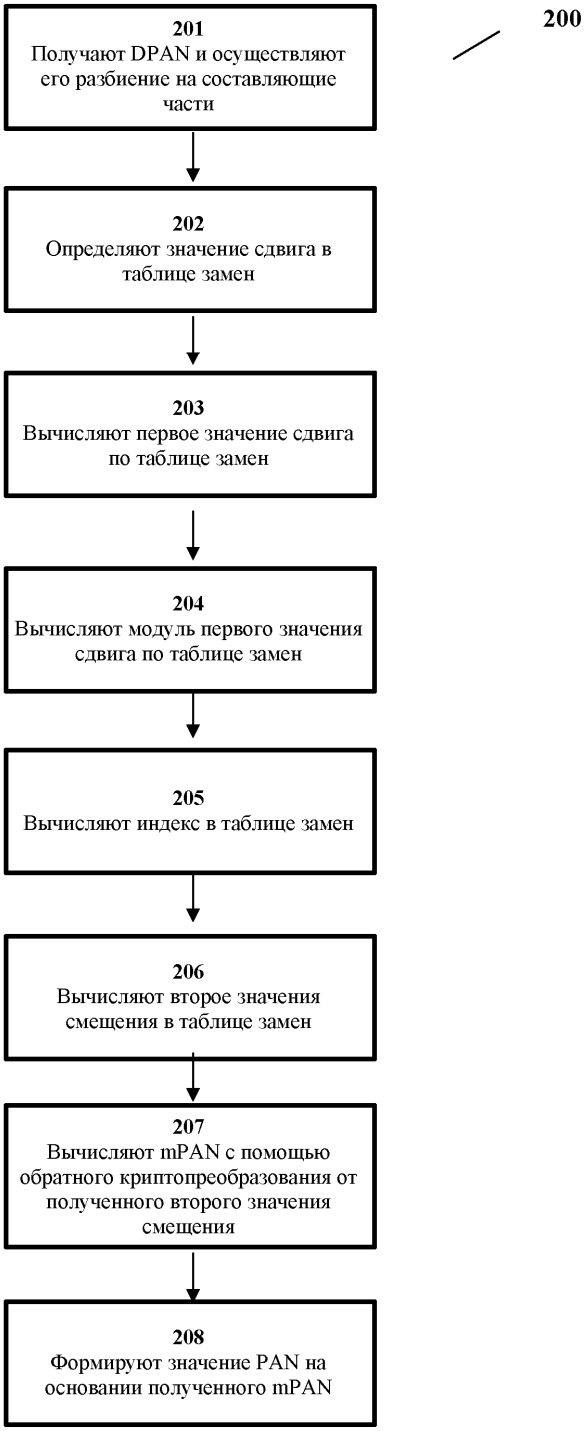
5. Способ по п.3, в котором если второе значение смещения отрицательное, то итеративно повторяют шаг его вычисления, инкрементировав модуль первого значения смещения на единицу.

6. Система для токенизации номера банковской карты (PAN), содержащая по меньшей мере один процессор и по меньшей мере одно средство хранения данных, содержащее машиночитаемые инструкции, которые при их выполнении процессором реализуют способ по любому из пп.1, 2.

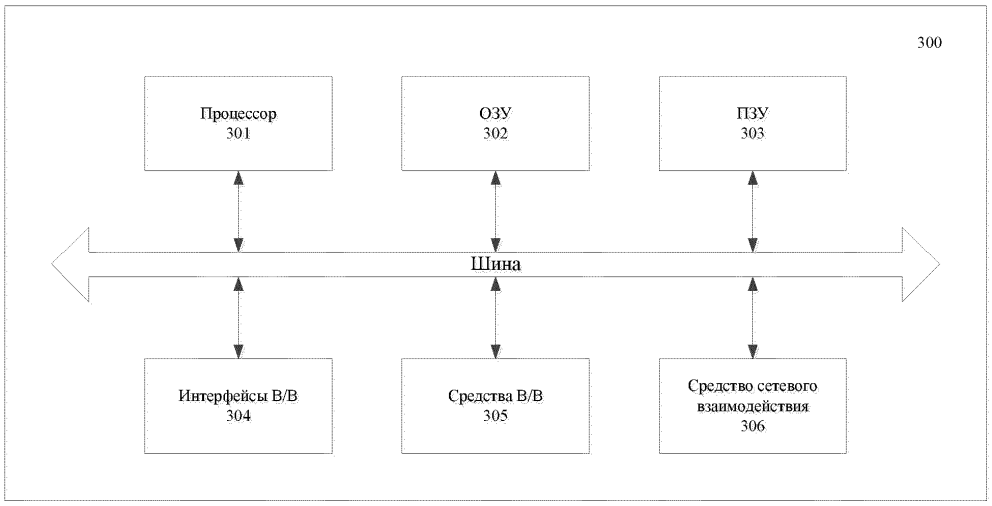
7. Система для определения детокенизированного номера банковской карты (DPAN), полученного с помощью способа по любому из пп.1, 2, содержащая по меньшей мере один процессор и по меньшей мере одно средство хранения данных, содержащее машиночитаемые инструкции, которые при их выполнении процессором реализуют способ по любому из пп.3-5.



Фиг. 1


200
201
Получают DPAN и осуществляют его разбиение на составляющие части
202
Определяют значение сдвига в таблице замен
203
Вычисляют первое значение сдвига по таблице замен
204
Вычисляют модуль первого значения сдвига по таблице замен
205
Вычисляют индекс в таблице замен
206
Вычисляют второе значения смещения в таблице замен
207
Вычисляют mPAN с помощью обратного криптопреобразования от полученного второго значения смещения
208
Формируют значение PAN на основании полученного mPAN

Фиг. 2


300
Процессор
301
ОЗУ
302
ПЗУ
303
Шина
Интерфейсы В/В
304
Средства В/В
305
Средство сетевого
взаимодействия
306

Фиг. 3