РОССИЙСКАЯ ФЕДЕРАЦИЯ

(19) **RU** (11) **2 638 639**(13) **C1**

(51) МПК
H04L 9/06 (2006.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: **2017105340, 07.08.2015**

(24) Дата начала отсчета срока действия патента:
07.08.2015

Дата регистрации:
14.12.2017

Приоритет(ы):
(30) Конвенционный приоритет:
07.08.2014 GB 1414007.3

(45) Опубликовано: **14.12.2017** Бюл. № **35**

(85) Дата начала рассмотрения заявки PCT на национальной фазе: **07.03.2017**

(86) Заявка PCT:
EP 2015/025056 (07.08.2015)

(87) Публикация заявки PCT:
WO 2016/020068 (11.02.2016)

Адрес для переписки:
191036, Санкт-Петербург, а/я 24, "НЕВИНПАТ"

(72) Автор(ы):
КЯРККЯЙНЕН Туомас (FI)

(73) Патентообладатель(и):
Гуруподжик Микросистемс Ой (FI)

(56) Список документов, цитированных в отчете о поиске: **US 2006/0056625 A1, 16.03.2006. EP 1209844 A2, 29.05.2002. EP 1037131 A2, 20.09.2000. US 7076064 B2, 11.07.2006. US 2002/0066012 A1, 30.05.2002. RU 2273894 C2, 10.04.2006.**

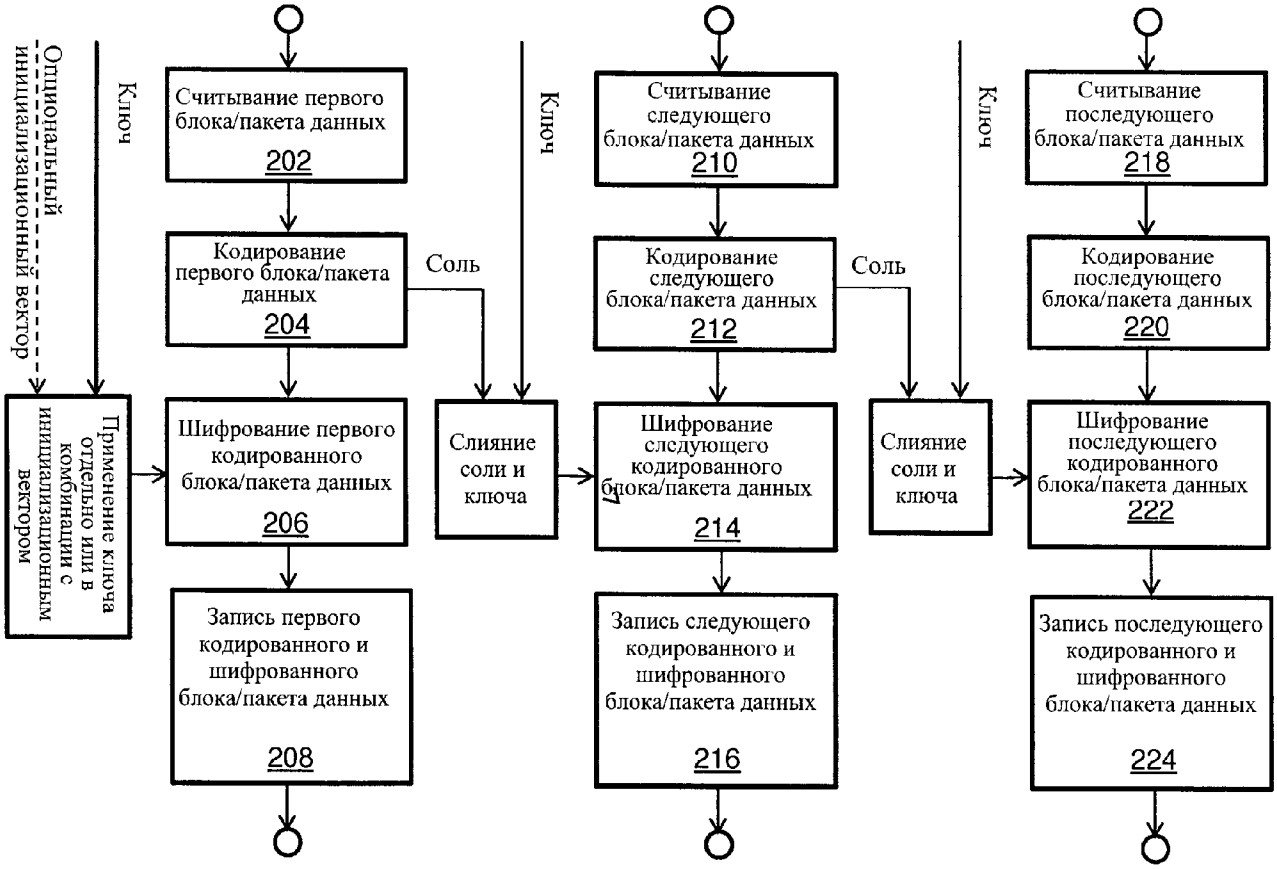
(54) Кодер, декодер и способ кодирования и шифрования входных данных

(57) Реферат:

Изобретение относится к средствам кодирования и шифрования входных данных для формирования соответствующих кодированных и шифрованных данных. Технический результат заключается в расширении арсенала технических средств кодирования данных. Кодируют по меньшей мере первый блок данных из входных данных, в результате чего формируют первый кодированный блок данных. Первый кодированный блок данных затем шифруют с использованием по меньшей мере одного ключа, чтобы получить первый шифрованный и кодированный блок данных, который включают в кодированные и шифрованные данные. При этом также формируют первое начальное значение для использования при шифровании следующего кодированного блока, чтобы получить следующий кодированный и шифрованный блок данных, который включают в кодированные и шифрованные данные. Далее формируют следующее начальное значение для использования при шифровании последующего кодированного блока данных, и так далее последовательно и повторно до тех пор, пока все блоки данных входных данных не будут кодированы и шифрованы в виде кодированных и шифрованных данных. 6 н. и 36 з.п. ф-лы, 5 ил.

RU 2638639 C1

RU 2638639 C1



Фиг. 2

RU 2638639 C1

RU 2638639 C1

RUSSIAN FEDERATION

FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(19) **RU** (11) **2 638 639**(13) **C1**

(51) Int. Cl.
H04L 9/06 (2006.01)

(12) ABSTRACT OF INVENTION

(21)(22) Application: **2017105340, 07.08.2015**

(24) Effective date for property rights:
07.08.2015

Registration date:
14.12.2017

Priority:
(30) Convention priority:
07.08.2014 GB 1414007.3

(45) Date of publication: **14.12.2017** Bull. № **35**

(85) Commencement of national phase: **07.03.2017**

(86) PCT application:
EP 2015/025056 (07.08.2015)

(87) PCT publication:
WO 2016/020068 (11.02.2016)

Mail address:
191036, Sankt-Peterburg, a/ya 24, "NEVINPAT"

(72) Inventor(s):
KARKKAINEN Tuomas (FI)

(73) Proprietor(s):
Gurulogic Microsystems Oy (FI)

RU 2638639 C1

(54) ENCODER, DECODER AND METHOD FOR ENCODING AND ENCRYPTING INPUT DATA

(57) Abstract:

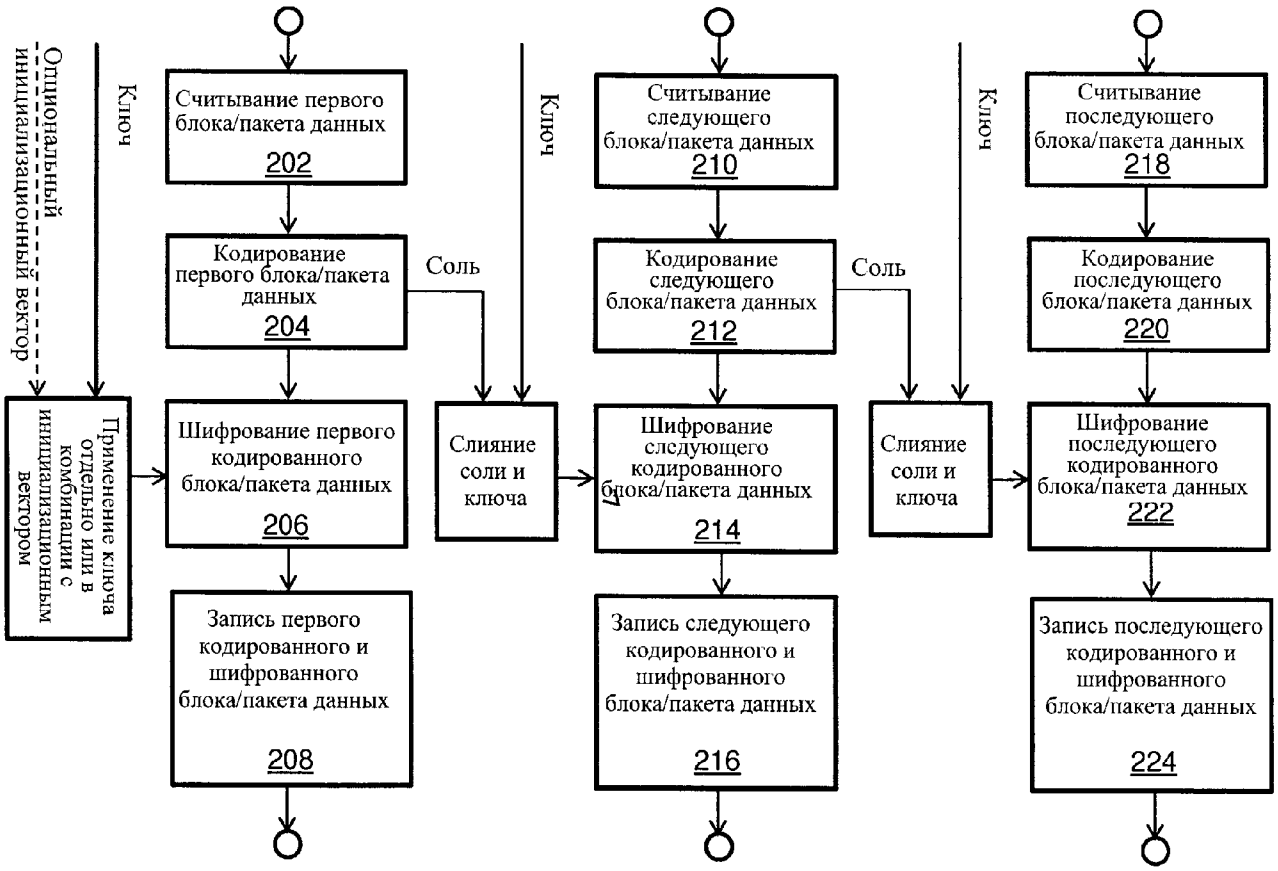
FIELD: physics.

SUBSTANCE: at least the first data unit is encoded from the input data, thereby forming the first encoded data unit. The first encoded data unit is then encrypted using at least one key to obtain the first encrypted and encoded data unit which is included in the encoded and encrypted data. The first initial value for use in encryption of the next encoded unit is also generated to obtain the next encoded and encrypted data unit which is included in the encoded and encrypted data. Further, the next initial value is generated for use in encrypting the subsequent encoded data unit, and so on sequentially and repeatedly until all data units of the input data are encoded and encrypted as encoded and encrypted data.

EFFECT: extending the range of technical facilites of data coding.

42 cl, 5 dwg

RU 2638639 C1



Фиг. 2

RU 2638639 C1

RU 2638639 C1

Область техники

Настоящее изобретение относится к кодерам для кодирования и шифрования входных данных (D1) с целью формирования соответствующих кодированных и шифрованных данных (E2), а также к соответствующим способам кодирования и шифрования входных данных (D1) с целью формирования соответствующих кодированных и шифрованных данных (E2). Также настоящее изобретение относится к декодерам для дешифрования и декодирования кодированных и шифрованных данных (E2) с целью формирования соответствующих декодированных данных (D3), а также к соответствующим способам дешифрования и декодирования кодированных и шифрованных данных (E2) с целью формирования соответствующих декодированных данных (D3). Также настоящее изобретение относится к компьютерным программным продуктам, включающим машиночитаемый носитель, на котором хранят машиночитаемые инструкции, при этом машиночитаемые инструкции могут быть исполнены на компьютеризованном устройстве, включающем процессорную аппаратуру для исполнения упомянутых выше способов. Настоящее изобретение относится также к кодекам, включающим по меньшей мере один упомянутый кодер и по меньшей мере один упомянутый декодер.

Предпосылки создания изобретения

В общем случае под «шифрованием» понимают процедуру кодирования сообщений или информации таким образом, при котором прочитать эти сообщения или информацию способны только авторизованные стороны. Область знаний, которая занимается шифрованием, называют криптографией. Шифрование информации применялось на всем протяжении человеческой истории, и хорошо известно, что любой алгоритм шифрования имеет собственные уязвимости. Криптоанализ - это раздел криптографии, который применяют для поиска уязвимостей в алгоритмах шифрования.

Алгоритмы шифрования делятся на два класса: симметричные алгоритмы (т.е. алгоритмы с симметричным ключом) и ассиметричные алгоритмы (т.е. алгоритмы с асимметричными ключами). Симметричные и асимметричные алгоритмы отличаются друг от друга в том, каким образом используют и обрабатывают ключ шифрования. В симметричных алгоритмах шифрования для шифрования данных на передающем конце и для дешифрования данных на соответствующем приемном конце используют разделяемый общий ключ. В отличие от симметричных алгоритмов, в асимметричных алгоритмах шифрования используют два различных ключа, один из которых - публичный ключ, применяют для шифрования данных, а второй - приватный ключ, применяют для дешифрования шифрованных данных. Стороны связи разделяют только публичный ключ.

При этом существуют также односторонние функции свертывания сообщений, то есть хэш-функции, которые сами по себе не относятся к методам шифрования, поскольку формируемые ими данные очень сложно или вообще невозможно восстановить. Однако односторонние функции свертки сообщений могут быть использованы для проверки подлинности данных и паролей, а также в целях формирования ключей шифрования для алгоритмов шифрования.

Общеизвестно, что операции шифрования данных предъявляют высокие требования к аппаратному обеспечению и требуют большого объема вычислительных ресурсов. Соответственно, в целях экономии вычислительных ресурсов и сокращения времени вычислений, часто применяют гибридные комбинации из ассиметричных и симметричных алгоритмов шифрования. Они позволяют получить достаточно высокую степень защиты, гарантирующую, что никакая третья сторона, обладая современными вычислительными ресурсами, не сможет выполнять несанкционированное дешифрование

защищенной информации в реальном времени. Такой подход чаще всего используют в различных протоколах передачи данных, например, в протоколе защищенных сокетов (Secure Sockets Layer (SSL)) / безопасности транспортного уровня (Transport Layer Security, TLS) и протоколе безопасной оболочки (Secure Shell (SSH)), а также в приложениях для подписи и шифрования сообщений электронной почты, таких, например, как приложение Pretty Good Privacy (PGP) [«весьма хорошая конфиденциальность»]

В современных условиях криптология, то есть научное изучение криптографии и криптоанализа, является непрерывно развивающейся областью знания, что в случае криптоанализа означает непрерывные попытки поиска уязвимостей в алгоритмах шифрования. По этой причине крайне важно обеспечивать максимальную защиту информации, однако вместе с тем всегда необходимы компромиссы в отношении вычислительных ресурсов, требуемых для реализации шифрования. При этом доступные вычислительные ресурсы, как правило, ограничены, особенно в мобильных устройствах, где предпринимают все возможные меры для сбережения энергии аккумулятора.

В заявке на патент США 2006/0188095 A1 «Комбинированный способ кодирования для одновременного шифрования и кодирования канала, передающее устройство к нему, комбинированный способ декодирования для одновременного декодирования канала и дешифрования, и приемное устройство для него» описан способ комбинированного кодирования, передающее устройство для него, способ комбинированного декодирования и приемное устройство для него. Передающее устройство, при помощи которого выполняют одновременное шифрование и кодирование канала, включает блок комбинированного кодирования для выполнения комбинированного кодирования над исходным кодированным сообщением и для вывода комбинированно кодированного сообщения. Приемное устройство, при помощи которого выполняют одновременное декодирование канала и дешифрование комбинированно кодированного сообщения с добавленным шумом, включает блок комбинированного декодирования для выполнения комбинированного декодирования над комбинированно кодированным сообщением с добавленным шумом из демодулятора и для вывода исходного кодированного сообщения.

В патенте США 8660261 B2 «Система и устройство для интегрированного кодирования/декодирования и шифрования/дешифрования видеоданных и данных изображений» описан энтропийный кодер с функциональностью шифрования, предназначенный для применения в мультимедийном кодеке. В энтропийном кодере реализуют рандомизированную схему кодирования Хаффмана, без хранения множества наборов таблиц Хаффмана в постоянной памяти (read-only memory, ROM). Энтропийный кодер включает память ROM, в которой хранят один набор таблиц кодирования, кодер включает также блок поиска по таблице, который связан с ROM-памятью и который используют для преобразования символов в исходные кодовые слова и наоборот при помощи поиска по таблице, и блок рандомизатора таблицы, которую используют для преобразования исходных кодовых слов, кодированных методом Хаффмана, в кодовые слова рандомизированного кода Хаффмана и наоборот при помощи алгоритма формирования изоморфного кода. Блок рандомизации таблицы обеспечивает преобразование на основе ключевой последовательности скачков, формируемой при помощи генератора псевдослучайных битов с использованием ключа шифрования/дешифрования.

В заявке на патент США 2006/0056625A1 описан способ шифрования данных с использованием последовательности случайных чисел, генерируемой блоком генерирования случайных чисел для генерирования последовательности случайных

чисел, уникально определяемой на основании входного параметра, при этом способ включает шаг генерирования входного параметра на основе метаданных шифруемых данных.

В опубликованной статье Samarakoon et al.; "Encrypted video over TETRA" описана система мобильной связи с улучшенной защитой. Система использует сквозное шифрование в дополнение к шифрованию радиосигнала. Система использует вставку кадров для обеспечения сквозного шифрования. Кадры вставляют в передаваемый видеопоток между последовательными видеокадрами во избежание потерь данных. Однако, чтобы позволить выполнить вставку, приложение должно уменьшить скорость данных для поддержания той же самой общей скорости передачи.

Сущность изобретения

Цель настоящее изобретения - предложить кодер для кодирования и шифрования входных данных (D1) с целью формирования соответствующих кодированных и шифрованных данных (E2).

Также цель настоящего изобретения - предложить усовершенствованный декодер для дешифрования и декодирования кодированных и шифрованных данных (E2) с целью формирования соответствующих декодированных данных (D3).

В первом аспекте вариантов осуществления настоящего изобретения предложен кодер для кодирования и шифрования входных данных (D1), включающих множество блоков данных или пакетов данных или потоков данных, при этом кодер (110) включает схему обработки данных для обработки входных данных (D1) для формирования соответствующих кодированных и шифрованных данных (E2), при этом упомянутая схема обработки данных объединяет процессы кодирования и шифрования для формирования кодированных и шифрованных данных (E2), причем:

(i) упомянутая схема обработки данных выполнена с возможностью кодирования по меньшей мере первого блока данных или пакета данных или потока данных из упомянутого множества блоков данных или пакетов данных или потоков данных для формирования первого кодированного блока данных или пакета данных или потока данных, и шифрования упомянутого по меньшей мере первого кодированного блока данных или пакета данных или потока данных с использованием по меньшей мере одного ключа, чтобы получить первый кодированный и шифрованный блок данных или пакет данных или поток данных для включения в кодированные и шифрованные данные (E2);

(ii) упомянутая схема обработки данных выполнена с возможностью формирования первого начального значения для использования при шифровании следующего кодированного блока данных или пакета данных или потока данных, чтобы получить следующий кодированный и шифрованный блок данных или пакет данных или поток данных для включения в кодированные и шифрованные данные (E2);

(iii) упомянутая схема обработки данных выполнена с возможностью формирования следующего начального значения для использования при шифровании последующего кодированного блока данных или пакета данных или потока данных, и так последовательно и повторно до тех пор, пока упомянутое множество блоков данных или пакетов данных или потоков данных не будет шифровано и кодировано в виде кодированных и шифрованных данных (E2),

при этом для данного кодируемого и шифруемого блока данных или пакета данных или потока данных начальное значение формируют на основании предшествующего ему блока данных или пакета данных или потока данных.

Настоящее изобретение обладает тем преимуществом, что оно позволяет получить

усовершенствованный тип кодера для кодирования данных с целью формирования соответствующих кодированных данных с применением начальных значений.

При этом, опционально, упомянутая схема обработки данных выполнена с возможностью кодирования и шифрования входных данных (D1) с целью формирования соответствующих кодированных и шифрованных данных (E2), последовательно и повторно, с ветвлением на множество параллельных последовательностей операций по кодированию и шифрованию блоков данных или пакетов данных или потоков данных с использованием соответствующих начальных значений.

Опционально, упомянутую схему обработки данных в кодере реализуют с применением по меньшей мере одного процессора с архитектурой с ограниченным набором команд (Reduced Instruction Set Computing, RISC), который выполнен с возможностью исполнения программных инструкций, в соответствии с последующим более подробным описанием.

Опционально, упомянутая схема обработки данных в кодере выполнена с возможностью кодирования и шифрования входных данных (D1), представленных в по меньшей мере одной из следующих форм: одномерные данные, многомерные данные, текстовые данные, двоичные данные, данные датчиков, аудиоданные, данные изображение, видеоданные, однако без ограничения перечисленным.

Опционально, схему обработки данных в кодере, в процессе ее функционирования, снабжают упомянутым по меньшей мере одним ключом, который используют при формировании кодированных и шифрованных данных (E2).

Опционально, упомянутая схема обработки данных в кодере выполнена с возможностью многократного использования упомянутого по меньшей мере одного ключа в комбинации с упомянутыми начальными значениями для шифрования кодированных блоков данных или пакетов данных или потоков данных с целью их включения в кодированные и шифрованные данные (E2). Альтернативно и опционально, упомянутая схема обработки данных в кодере выполнена с возможностью использования упомянутого по меньшей мере одного ключа для шифрования только упомянутого первого кодированного блока данных и/или пакета данных.

Альтернативно и опционально, упомянутая схема обработки данных в кодере выполнена с возможностью применения инициализационного вектора (IV) в комбинации с упомянутым по меньшей мере одним ключом для шифрования упомянутого первого кодированного блока данных или пакета данных или потока данных.

Также, опционально, упомянутая схема обработки данных в кодере выполнена с возможностью включения в кодированные и шифрованные данные (E2) информации, указывающей по меньшей один алгоритм, использованный для формирования начальных значений с целью их использования при шифровании кодированных блоков данных или пакетов данных или потоков данных.

Также, опционально, упомянутая схема обработки данных в кодере выполнена с возможностью обеспечения передачи упомянутого по меньшей мере одного ключа для использования при последующем дешифровании и декодировании кодированных и шифрованных данных (E2), вручную или при помощи шифрованного сообщения электронной почты, или по шифрованному соединению связи. Опционально, упомянутое шифрованное соединение связи реализуют при помощи протокола защищенных сокетов (SSL) / безопасности транспортного уровня (TLS).

Во втором аспекте вариантов осуществления настоящего изобретения предложен способ кодирования и шифрования входных данных (D1), включающих множество блоков данных или пакетов данных или потоков данных, при помощи кодера (110),

который включает схему обработки данных для обработки входных данных (D1) для формирования соответствующих кодированных и шифрованных данных (E2), при этом упомянутая схема обработки данных объединяет процессы кодирования и шифрования для формирования кодированных и шифрованных данных (E2), и способ включает:

(i) кодирование первого блока данных или пакета данных или потока данных из упомянутого множества блока данных или пакета данных или потока данных для формирования по меньшей мере первого кодированного блока данных или пакета данных или потока данных;

(ii) шифрование упомянутого по меньшей мере первого кодированного блока данных или пакета данных или потока данных с использованием по меньшей мере одного ключа, чтобы получить первый шифрованный и кодированный блок данных или пакет данных или поток данных для включения в кодированные и шифрованные данные (E2);

(iii) формирование первого начального значения для использования при шифровании следующего кодированного блока данных или пакета данных или потока данных, чтобы получить следующий кодированный и шифрованный блок данных или пакет данных или поток данных для включения в кодированные и шифрованные данные (E2); и

(iv) формирование следующего начального значения для использования при шифровании последующего кодированного блока данных или пакета данных или потока данных, и так последовательно и повторно до тех пор, пока упомянутое множество блоков данных или пакетов данных или потоков данных не будет шифровано и кодировано в виде кодированных и шифрованных данных (E2),

при этом для данного кодируемого и шифруемого блока данных или пакета данных или потока данных начальное значение формируют на основании предшествующего ему блока данных или пакета данных или потока данных.

Опционально способ включает предоставление, в упомянутую схему обработки данных, по меньшей мере одного ключа, который используют при формировании кодированных и шифрованных данных (E2).

Опционально способ включает многократное использование упомянутой схемой обработки данных упомянутого по меньшей мере одного ключа в комбинации с упомянутыми начальными значениями для шифрования кодированных блоков данных или пакетов данных или потоков данных для их включения в кодированные и шифрованные данные (E2).

Опционально способ включает применение упомянутой схемой обработки данных упомянутого по меньшей мере одного ключа для шифрования только упомянутого первого кодированного блока данных или пакета данных или потока данных.

Опционально способ включает применение, упомянутой схемой обработки данных, инициализационного вектора (IV) в комбинации с упомянутым по меньшей мере одним ключом при шифровании упомянутого первого кодированного блока данных или пакета данных или потока данных.

Опционально способ включает выполнение, упомянутой схемой обработки данных, включения в кодированные и шифрованные данные (E2) информации, указывающей по меньшей мере один алгоритм, примененный для формирования начальных значений для использования при шифровании кодированных блоков данных или пакетов данных или потоков данных.

Опционально способ включает выполнение, упомянутой схемой обработки данных, кодирования и шифрования входных данных (D1) для формирования соответствующих

кодированных и шифрованных данных (E2), последовательно и повторно, с ветвлением на множество параллельных последовательностей операций по кодированию и шифрованию блоков данных или пакетов данных или потоков данных с использованием соответствующих начальных значений.

Опционально способ включает выполнение, упомянутой схемой обработки данных, кодирования и шифрования входных данных (D1), представленных по меньшей мере в одной из следующих форм: одномерные данные, многомерные данные, текстовые данные, двоичные данные, данные датчиков, аудиоданные, данные изображение, видеоданные.

Опционально способ включает выполнение, упомянутой схемой обработки данных, передачи упомянутого по меньшей мере одного ключа из кодера (110) для использования при последующем дешифровании и декодировании кодированных и шифрованных данных (E2), вручную или при помощи шифрованного сообщения электронной почты, или по шифрованному соединению связи. Также опционально способ включает реализацию упомянутого соединения связи при помощи протокола защищенных сокетов (SSL) / безопасности транспортного уровня (TLS).

Опционально способ включает реализацию упомянутой схемы обработки данных с применением по меньшей мере одного процессора с архитектурой с ограниченным набором команд (RISC), который выполнен с возможностью исполнения программных инструкций.

В третьем аспекте вариантов осуществления настоящего изобретения предложен компьютерный программный продукт, включающий машиночитаемый носитель, на котором хранят машиночитаемые инструкции, при этом машиночитаемые инструкции могут быть исполнены на компьютеризованном устройстве, включающем процессорную аппаратуру для исполнения описанного выше способа.

В четвертом аспекте вариантов осуществления настоящего изобретения предложен декодер для дешифрования и декодирования кодированных и шифрованных данных (E2), включающих множество кодированных и шифрованных блоков данных или пакетов данных или потоков данных, при этом декодер включает схему обработки данных для обработки кодированных и шифрованных данных (E2) для формирования соответствующих декодированных данных (D3), и в декодер, при его функционировании, предоставляют по меньшей мере один ключ, который используют при формировании декодированных данных (D3), при этом упомянутая схема обработки данных объединяет процессы декодирования и дешифрования для формирования декодированных данных (D3), причем:

(i) упомянутая схема обработки данных выполнена с возможностью дешифрования по меньшей мере первого кодированного и шифрованного блока данных или пакета данных или потока данных из упомянутого множества кодированных и шифрованных блоков данных или пакетов данных или потоков данных с использованием упомянутого по меньшей мере одного ключа, чтобы получить по меньшей мере первый кодированный блок данных или пакет данных или поток данных, и декодирования упомянутого по меньшей мере первого кодированного блока данных или пакета данных или потока данных, чтобы получить первый декодированный блок данных или пакет данных или поток данных для включения в декодированные данные (D3);

(ii) упомянутая схема обработки данных выполнена с возможностью формирования первого начального значения для использования при дешифровании по меньшей мере следующего кодированного и шифрованного блока данных или пакета данных или потока данных из упомянутого множества кодированных и шифрованных блоков

данных или пакетов данных или потоков данных, чтобы получить по меньшей мере следующий кодированный блок данных или пакет данных или поток данных, и декодирования упомянутого следующего кодированного блока данных или пакета данных или потока данных, чтобы получить следующий декодированный блок данных или пакет данных или поток данных для включения в декодированные данные (D3); и

(iii) упомянутая схема обработки данных выполнена с возможностью формирования следующего начального значения для использования при дешифровании и декодировании последующего кодированного и шифрованного блока данных или пакета данных или потока данных из упомянутого множества кодированных и шифрованных блоков данных или пакетов данных или потоков данных, и так последовательно и повторно до тех пор, пока упомянутое множество кодированных и шифрованных блоков данных или пакетов данных или потоков данных не будет дешифровано и декодировано в декодированные данные (D3),

при этом для данного кодированного и шифрованного блока данных или пакета данных или потока данных начальное значение формируют на основании предшествующего ему декодированного блока данных или пакета данных или потока данных.

При этом, опционально, упомянутая схема обработки данных выполнена с возможностью дешифрования кодированных и шифрованных данных (E2) с целью формирования соответствующих декодированных данных (D3), последовательно и повторно, с ветвлением на множество параллельных последовательностей операций по декодированию и дешифрованию шифрованных и кодированных блоков данных или пакетов данных или потоков данных с использованием соответствующих начальных значений.

Опционально, упомянутую схему обработки данных в декодере реализуют с применением по меньшей мере одного процессора с архитектурой с ограниченным набором команд (RISC), который выполнен с возможностью исполнения программных инструкций, в соответствии с последующим более подробным описанием; при этом RISC-процессор способен выполнять сравнительно простые операции конкатенации данных с очень высокой скоростью, и хорошо подходит для кодирования и декодирования данных, поступающих в потоковом формате, например, в реальном времени.

Опционально, упомянутая схема обработки данных в декодере выполнена с возможностью дешифрования и декодирования кодированных и шифрованных данных (E2), представленных в по меньшей мере одной из следующих форм: кодированные и шифрованные одномерные данные, кодированные и шифрованные многомерные данные, кодированные и шифрованные текстовые данные, кодированные и шифрованные двоичные данные, кодированные и шифрованные данные датчиков, кодированные и шифрованные аудиоданные, кодированные и шифрованные данные изображений, кодированные и шифрованные видеоданные, однако без ограничения перечисленным.

Опционально, в упомянутую схему обработки данных в декодере, в процессе ее функционирования, предоставляют упомянутый по меньшей мере один ключ, который используют при формировании декодированных данных (D3).

Опционально, упомянутая схема обработки данных выполнена с возможностью многократного использования упомянутого по меньшей мере одного ключа, в комбинации с упомянутыми начальными значениями для дешифрования множества кодированных и шифрованных блоков данных или пакетов данных или потоков данных.

Альтернативно и опционально, упомянутая схема обработки данных выполнена с возможностью использования упомянутого по меньшей мере одного ключа для дешифрования только упомянутого первого кодированного и шифрованного блока данных или пакета данных или потока данных.

Кроме того, опционально, упомянутая схема обработки данных выполнена с возможностью применения инициализационного вектора (IV) в комбинации с упомянутым по меньшей мере одним ключом для дешифрования упомянутого первого кодированного и шифрованного блока данных или пакета данных или потока данных.

Опционально, упомянутая схема обработки данных выполнена с возможностью обеспечения приема упомянутого по меньшей мере одного ключа в декодере для использования при последующем дешифровании и декодировании кодированных и шифрованных данных (E2), вручную или при помощи шифрованного сообщения электронной почты, или по шифрованному соединению связи. Также опционально, упомянутое шифрованное соединение связи реализовано при помощи протокола защищенных сокетов (SSL) / безопасности транспортного уровня (TLS).

В пятом аспекте вариантов осуществления настоящего изобретения предложен способ дешифрования и декодирования кодированных и шифрованных данных (E2), включающих множество кодированных и шифрованных блоков данных или пакетов данных или потоков данных, при помощи декодера, при этом декодер включает схему обработки данных для обработки кодированных и шифрованных данных (E2) с целью формирования соответствующих декодированных данных (D3), и в декодер, при его функционировании, предоставляют по меньшей мере один ключ, который используют при формировании декодированных данных (D3), при этом упомянутая схема обработки данных объединяет процессы декодирования и дешифрования для формирования декодированных данных (D3), и способ включает:

(i) дешифрование по меньшей мере первого кодированного и шифрованного блока данных или пакета данных или потока данных из упомянутого множества кодированных и шифрованных блоков данных или пакетов данных или потоков данных с использованием упомянутого по меньшей мере одного ключа, чтобы сформировать по меньшей мере первый кодированный блок данных или пакет данных или поток данных;

(ii) декодирование упомянутого по меньшей мере первого блока данных или пакета данных или потока данных, чтобы получить первый декодированный блок данных или пакет данных или поток данных для включения в декодированные данные (D3);

(iii) формирование первого начального значения для использования при дешифровании и декодировании следующего кодированного и шифрованного блока данных или пакета данных или потока данных из упомянутого множества кодированных и шифрованных блоков данных или пакетов данных или потоков данных, чтобы получить по меньшей мере следующий декодированный блок данных или пакет данных или поток данных для включения в декодированные данные (D3); и

(iv) формирование следующего начального значения для использования при дешифровании и декодировании последующего кодированного и шифрованного блока данных или пакета данных или потока данных из упомянутого множества кодированных и шифрованных блоков данных или пакетов данных или потоков данных, и так последовательно и повторно до тех пор, пока упомянутое множество кодированных и шифрованных блоков данных или пакетов данных или потоков данных не будет дешифровано и декодировано в декодированные данные (D3),

при этом для данного кодированного и шифрованного блока данных или пакета

данных или потока данных начальное значение формируют на основании предшествующего ему декодированного блока данных или пакета данных или потока данных.

Опционально способ включает многократное использование упомянутой схемой обработки данных упомянутого по меньшей мере одного ключа в комбинации с упомянутыми начальными значениями для дешифрования упомянутого множества кодированных и шифрованных блоков данных или пакетов данных или потоков данных.

Опционально способ включает использование упомянутой схемой обработки данных упомянутого по меньшей мере одного ключа для дешифрования только упомянутого первого кодированного и шифрованного блока данных или пакета данных или потока данных.

Опционально способ включает выполнение упомянутой схемы обработки данных с возможностью применения инициализационного вектора (IV) в комбинации с упомянутым по меньшей мере одним ключом при дешифровании упомянутого по меньшей мере первого кодированного и шифрованного блока данных или пакета данных или потока данных.

Опционально способ включает выполнение, упомянутой схемой обработки данных, дешифрования кодированных и шифрованных данных (E2) для формирования соответствующих декодированных данных (D3), последовательно и повторно, с ветвлением на множество параллельных последовательностей операций по декодированию и дешифрованию шифрованных и кодированных блоков данных или пакетов данных или потоков данных с использованием соответствующих начальных значений.

Опционально способ включает выполнение, упомянутой схемой обработки данных, дешифрования и декодирования кодированных и шифрованных данных (E2), представленных по меньшей мере в одной из следующих форм: кодированные и шифрованные одномерные данные, кодированные и шифрованные многомерные данные, кодированные и шифрованные текстовые данные, кодированные и шифрованные двоичные данные, кодированные и шифрованные данные датчиков, кодированные и шифрованные аудиоданные, кодированные и шифрованные данные изображений, кодированные и шифрованные видеоданные.

Опционально способ включает выполнение, упомянутой схемой обработки данных, приема упомянутого по меньшей мере одного ключа в декодере для использования при последующем дешифровании и декодировании кодированных и шифрованных данных (E2), вручную или при помощи шифрованного сообщения электронной почты, или по шифрованному соединению связи. Также опционально, упомянутое шифрованное соединение связи реализуют при помощи протокола защищенных сокетов (SSL) / безопасности транспортного уровня (TLS).

Опционально, упомянутую схему обработки данных реализуют с применением по меньшей мере одного процессора с архитектурой с ограниченным набором команд (RISC), который выполнен с возможностью исполнения программных инструкций.

В шестом аспекте вариантов осуществления настоящего изобретения предложен машиночитаемый носитель, на котором хранят машиночитаемые инструкции, при этом машиночитаемые инструкции могут быть исполнены на компьютеризованном устройстве, включающем процессорную аппаратуру для исполнения описанного выше способа.

В седьмом аспекте вариантов осуществления настоящего изобретения предложен кодек, включающий упомянутый кодер и упомянутый декодер.

Описанные выше способы обеспечивают значительное усиление защиты по сравнению с защитой, обеспечиваемой при помощи способов, известных на существующем уровне техники, в которых применяют соответствующие алгоритмы шифрования. Способы, предложенные в вариантах осуществления настоящего изобретения, могут быть реализованы с использованием любых подходящих методов кодирования, независимо от применяемого алгоритма шифрования. Благодаря этому, предложенные способы не меняют поведения применяемых алгоритмов шифрования, и это означает, что эффективность защиты, обеспечиваемой встроенными алгоритмами шифрования, не будет снижена. Следовательно, способы, предложенные в вариантах осуществления настоящего изобретения, позволяют еще более повысить эффективность существующих алгоритмов сжатия данных и шифрования.

При этом предложенные способы могут быть реализованы в комбинации с распространенными общеизвестными программными приложениями для сжатия данных, как с открытым кодом, так и проприетарными, например, 7-Zip, Win-Zip и др. (наименования "7-Zip" и "Win-Zip" являются зарегистрированными товарными знаками).

При этом, также, интеграция процедур шифрования и кодирования является эффективной моделью для многопроцессорной обработки данных или для параллельного выполнения нескольких процедур. Интеграция процедур друг с другом позволяет реализовать оптимальную структуру обработки данных для каждого заданного центрального процессорного блока (Central Processing Unit, CPU) и заданного графического процессора, в соответствии с доступной вычислительной мощностью. То есть, описанные выше способы позволяют реализовать эффективное разделение на потоки для процедуры шифрования, встроенной в процедуру кодирования, при этом блоки данных и/или пакеты данных из входных данных (D1) могут быть представлены в оптимизированном формате, оптимальном для CPU и GPU системы и/или платформы, на которой выполняют процедуру кодирования.

Описанные выше способы позволяют применять очень быстрые, но вместе с тем в высокой степени эффективные алгоритмы шифрования. В этом отношении предложенные способы обеспечивают эффективное применение алгоритмов шифрования, без вмешательства в их внутреннее функционирование. Примеры алгоритмов шифрования, которые подходят для реализации в сочетании с упомянутыми выше способами включают, без ограничения перечисленным, AES, Twofish, Blowfish, стандарт шифрования данных (Data Encryption Standard, DES), Triple DES (3-DES), международный алгоритм шифрования данных (International Data Encryption Algorithm, IDEA), MARS, Rivest Cipher 6 (RC6), Camellia, CAST-128, Skipjack, расширенный миниатюрный алгоритм шифрования данных (extended Tiny Encryption Algorithm, XTEA) (перечисленные в качестве примеров наименования могут включать зарегистрированные товарные знаки).

При этом дополнительное преимущество от встраивания процедуры шифрования в процедуру кодирования заключается в том, что полученные таким образом кодированные и шифрованные данные (E2) не обязательно нужно передавать в другие сети с использованием защищенного, безопасного сетевого соединения, например, с применением тоннеля виртуальной частной сети (Virtual Private Network, VPN), протоколов безопасной оболочки или протоколов SSL7TLS. Следовательно, рассмотренные выше способы являются более выгодной моделью передачи текстовых, двоичных, аудиоданных, данных изображений, видеоданных и данных других типов, например, по публичным сетям Интернет или в веб-сервисах и облачных сервисах.

Для уяснения дополнительных аспектов, преимуществ, отличительных признаков и

целей настоящего изобретения следует обратиться к чертежам и подробному описанию примеров его осуществления, которые следует рассматривать в сочетании с приложенной формулой изобретения.

Нужно понимать, что отличительные признаки настоящего изобретения в пределах объема настоящего изобретения, заданного приложенной формулой изобретения, могут комбинироваться произвольным образом.

Описание чертежей

Краткое описание изобретения, приведенное выше, а также приведенное ниже подробное описание примеров осуществления настоящего изобретения, могут быть поняты более детально при его прочтении в сочетании с приложенными чертежами. В целях иллюстрации настоящего изобретения на чертежах показаны различные его примеры. Однако настоящее изобретение не ограничено конкретными способами и устройствами, описанными в данном документе. При этом специалисты в данной области техники должны понимать, что чертежи выполнены не в масштабе. Там, где это возможно, аналогичные элементы обозначены аналогичными числовыми обозначениями.

Далее, исключительно в качестве примера и со ссылками на приложенные чертежи, будут описаны примеры осуществления настоящего изобретения, где:

фиг. 1 представляет собой эскизную иллюстрацию кодера для кодирования и шифрования входных данных (D1) с целью формирования соответствующих кодированных и шифрованных данных (E2) и декодера для дешифрования и декодирования кодированных и шифрованных данных (E2) с целью формирования соответствующих декодированных данных (D3), при этом упомянутые кодер и декодер вместе образуют кодек, в соответствии с одним из вариантов осуществления настоящего изобретения;

фиг. 2 представляет собой эскизную блок-схему алгоритма, которая иллюстрирует шаги первого интегрированного способа кодирования и шифрования входных данных (D1) с целью формирования соответствующих кодированных и шифрованных данных (E2), в соответствии с одним из вариантов осуществления настоящего изобретения;

фиг. 3 представляет собой эскизную блок-схему алгоритма, которая иллюстрирует шаги второго интегрированного способа дешифрования и декодирования кодированных и шифрованных данных (E2) с целью формирования соответствующих декодированных данных (D3), в соответствии с одним из вариантов осуществления настоящего изобретения;

фиг. 4 представляет собой эскизную блок-схему алгоритма, которая иллюстрирует шаги третьего интегрированного способа кодирования и шифрования входных данных (D1) с целью формирования соответствующих кодированных и шифрованных данных (E2), в соответствии с еще одним из вариантов осуществления настоящего изобретения; и

фиг. 5 представляет собой эскизную блок-схему алгоритма, которая иллюстрирует шаги четвертого интегрированного способа дешифрования и декодирования кодированных и шифрованных данных (E2) с целью формирования соответствующих декодированных данных (D3), в соответствии с еще одним из вариантов осуществления настоящего изобретения.

На приложенных чертежах числа, выделенные подчеркиванием, используются для обозначения элементов, над которыми находится подчеркнутое число, или рядом с которыми оно расположено. Неподчеркнутые числовые обозначения относятся к объектам, указанным линией, которая соединяет неподчеркнутое число и объект. Если

число не выделено подчеркиванием и сопровождается связанной с ним стрелкой, это неподчеркнутое число используется для обозначения обобщенного элемента, на который указывает стрелка.

Подробное описание вариантов осуществления изобретения

В приведенном ниже подробном описании рассмотрены варианты осуществления настоящего изобретения и способы, которыми они могут быть реализованы. В настоящем документе описан вариант осуществления настоящего изобретения, который представляется авторам наилучшим, однако специалисты в данной области техники должны понимать, что возможны также другие варианты осуществления или практического применения настоящего изобретения.

В общем, варианты осуществления настоящего изобретения относятся к способу шифрования данных, встроенному в кодер, и, с соответствующими преобразованиями, к способу дешифрования данных, встроенному в декодер. Данные, кодированные и шифрованные при помощи кодера с использованием упомянутого интегрированного способа шифрования, не могут быть декодированы без применения аналогичного способа, встроенного в соответствующий декодер. В настоящем документе описана процедура шифрования, которая встроена в процедуру кодирования заданного применяемого кодера. Это позволяет значительно усилить соответствующее шифрование без ухудшения характеристик и снижения производительности алгоритма шифрования, применяемого в процедуре шифрования. В описанном выше встроенном способе шифрования, предпочтительно, применяют существующие алгоритмы шифрования. Встраивание процедуры шифрования, выполняют, опционально, в кодерах, где применяются блоки данных и/или пакеты данных и обрабатываются одномерные или многомерные текстовые данные, двоичные данные, аудиоданные, данные изображений, видеоданные или данные других типов.

Цель вариантов осуществления настоящего изобретения - предложить эффективный метод шифрования данных за счет понижения сложности алгоритмов шифрования, который бы позволял экономить вычислительные ресурсы и электроэнергию, расходуемую устройствами обработки данных. При этом, также, по сравнению с существующими решениями, при идентичных параметрах алгоритмов шифрования, варианты осуществления настоящего изобретения обеспечивают значительно усиленную защиту. Это означает, например, что для достижения повышенного уровня защиты данных не потребуется дополнительных вычислительных ресурсов.

Специалисты в данной области техники должны понимать, что в настоящем документе под «открытым текстом» понимается нешифрованная информация, а под «шифрованным текстом» понимается шифрованная информация.

В соответствии с иллюстрацией фиг. 1, варианты осуществления настоящего изобретения относятся:

(i) к кодеру 110 для кодирования и шифрования входных данных (D1) с целью формирования соответствующих кодированных и шифрованных данных (E2), а также к соответствующим способам кодирования и шифрования входных данных (D1) с целью формирования соответствующих кодированных и шифрованных данных (E2);

(ii) к декодеру 120 для дешифрования и декодирования кодированных и шифрованных данных (E2) с целью формирования соответствующих декодированных данных (D3), а также к соответствующим способам дешифрования и декодирования кодированных и шифрованных данных (E2) с целью формирования соответствующих декодированных данных (D3); и

(iii) к кодеку 130, включающему комбинацию из по меньшей мере одного кодера и

по меньшей мере одного декодера, а именно, комбинацию из кодера 110 и декодера 120.

Опционально, декодированные данные (D3) в точности аналогичны входным данным (D1), в случае кодирования без потерь. Альтернативно и опционально, декодированные данные (D3) приблизительно аналогичны входным данным (D1), в случае кодирования с потерями. Также, альтернативно и опционально, декодированные данные (D3) могут отличаться от входных данных (D1), например, вследствие преобразования, однако содержат по существу ту же информацию, которая присутствовала во входных данных (D1); например, из практических соображений, декодированные данные (D3) могут отличаться от входных данных (D1), если требуется переформатирование декодированных данных (D3), например, в целях совместимости с различными типами платформ связи, уровнями программного обеспечения, устройствами связи и т.п.

Кодер 110 включает схему обработки данных, которую используют для обработки входных данных (D1) с целью формирования соответствующих кодированных и шифрованных данных (E2). Опционально, схему обработки данных в кодере 110 реализуют с применением по меньшей мере одного процессора с архитектурой с ограниченным набором команд (RISC), который выполнен с возможностью исполнения программных инструкций, в соответствии с последующим более подробным описанием; при этом RISC-процессор способен выполнять сравнительно простые операции конкатенации данных с очень высокой скоростью, и хорошо подходит для кодирования и декодирования данных, поступающих в потоковом формате, например, в реальном времени.

Опционально, упомянутая схема обработки данных в кодере 110 выполнена с возможностью кодирования и шифрования входных данных (D1), представленных в по меньшей мере одной из следующих форм: одномерные данные, многомерные данные, текстовые данные, двоичные данные, данные датчиков, аудиоданные, данные изображение, видеоданные, однако без ограничения перечисленным. Входные данные (D1) включают множество блоков данных и/или пакетов данных. Опционально, входные данные (D1) могут быть приняты в виде потока данных или файла данных.

Опционально, в схему обработки данных в кодере 110, в процессе ее функционирования, предоставляют по меньшей мере один ключ, который используют при формировании кодированных и шифрованных данных (E2). Альтернативно и опционально, упомянутая схема обработки данных в кодере 110 выполнена с возможностью формирования упомянутого по меньшей мере одного ключа с использованием подходящего алгоритма формирования ключа.

Схема обработки данных в кодере 110 выполнена с возможностью кодирования первого блока данных и/или пакета данных из упомянутого множества блока данных и/или пакета данных с целью формирования по меньшей мере первого кодированного блока данных и/или пакета данных. Схема обработки данных в кодере 110 в этом случае выполнена с возможностью шифрования первого блока данных и/или пакета данных с использованием упомянутого по меньшей мере одного ключа, чтобы получить первый кодированный и шифрованный блок данных и/или пакет данных, который включают в кодированные и шифрованные данные (E2).

При этом, также, упомянутая схема обработки данных в кодере 110 выполнена с возможностью формирования первого начального значения (seed value) для использования при шифровании следующего кодированного блока данных и/или пакета данных, чтобы получить следующий кодированный и шифрованный блок данных и/или пакет данных, который включают в кодированные и шифрованные данные (E2).

Также схема обработки данных в кодере 110 выполнена с возможностью формирования следующего начального значения для использования при шифровании последующего кодированного блока данных и/или пакета данных, и так последовательно и повторно до тех пор, пока упомянутое множество блоков данных и/или пакетов данных не будет шифровано и кодировано в виде кодированных и шифрованных данных (E2). Опционально, упомянутое множество блоков данных и/или пакетов данных кодируют и шифруют один за другим, в несколько этапов, получая кодированные и шифрованные данные (E2), а именно, в форме множества кодированных и шифрованных блоков данных и/или пакетов данных. Нужно понимать, что техническая реализация кодера 110 может быть различной в зависимости от алгоритма кодирования и алгоритма шифрования, которые используют для выполнения процедур кодирования и шифрования соответственно. Тем не менее, в предложенном техническом решении процедуры кодирования и шифрования интегрированы друг с другом. Другими словами, нужно понимать, что выбранный алгоритм шифрования, предпочтительно, включают, то есть встраивают, в кодер 110.

Цель такого встраивания - формирование начальных значений для каждого блока данных и/или пакета данных, кодируемого и шифруемого на основе предшествующего ему блока данных и/или пакета данных. В результате, выполняемое впоследствии дешифрование каждого из шифрованных и кодированных блоков данных и/или пакетов данных зависит от дешифрования и декодирования предшествующего ему блока данных и/или пакета данных, последовательно и повторно, в соответствии с приведенным ниже более подробным описанием. Последовательное шифрование, выполняемое повторно с использованием упомянутых начальных значений, позволяет повысить эффективность алгоритма шифрования, поскольку взломщику, чтобы найти возможное решение для дешифрования и декодирования кодированных и шифрованных данных (E2), понадобится полностью воспроизвести функциональность как кодера 110, так и декодера 120.

В одной из альтернативных реализаций сначала может выполняться шифрование заданного блока данных и/или пакета данных, и затем он может кодироваться в кодированные и шифрованные данные (E2). Однако нужно понимать, что предпочтительным является кодирование блоков данных и/или пакетов данных до шифрования, поскольку во входных данных (последующие) блоки данных и/или пакеты данных, по сравнению с предшествующими им блоками данных и/или пакетами данных, имеют лишь незначительные изменения, а именно являются частичными или полными дубликатами предшествующих им блоков данных и/или пакетов данных. Соответственно, упомянутая схема обработки данных в кодере 110 выполнена с возможностью применения подходящего алгоритма дедупликации с целью кодирования множества блоков данных и/или пакетов данных из входных данных (D1) в множество кодированных блоков данных и/или пакетов данных. В качестве одного из примеров, для кодирования множества блоков данных и/или пакетов данных может применяться способ, описанный в документе GB 1411451.6. В качестве другого примера, для кодирования множества блоков данных и/или пакетов данных может применяться способ, описанный в документе GB 1411531.5.

При этом, опционально, упомянутая схема обработки данных в кодере 110 выполнена с возможностью выполнения дополнительной процедуры кодирования над множеством кодированных блоков данных и/или пакетов данных, до или после шифрования этого множества кодированных блоков данных и/или пакетов данных. С этой целью, опционально, упомянутая схема обработки данных в кодере 110 выполнена с

возможностью применения по меньшей мере одного из следующего: кодирования энтропийным модификатором, дельта-кодирования, О-дельта-кодирования, диапазонного кодирования 1u или 8u, кодирования длин серий (Run Length Encoding, RLE), скользящего кодирования длин серий (Split RLE, SRLE) и/или интерполирующего кодирования. В настоящем документе под «дельта кодированием» понимается способ хранения или передачи данных в форме разностей между последовательными значениями данных, вместо файлов целиком, тогда как под термином «О-дельта-кодирование» понимается разностный тип кодирования, основанный на циклическом обращении в двоичной системе счета, например, соответствующий описанию в патентном документе GB 1303661.1, который включен в настоящий документ путем ссылки. Под термином "SRLE-кодирование" или «скользящее RLE-кодирование» понимается способ скользящего кодирования длин серий, в соответствии с описанием в патентном документе GB 1303660.3. В одном из примеров, опционально, упомянутая схема обработки данных в кодере 110 выполнена с возможностью дальнейшего сжатия множества кодированных блоков данных и/или пакетов данных перед шифрованием за счет применения одного или более соответствующих способов кодирования с изменением энтропии. В качестве одного из примеров, предпочтительно, для кодирования применяют решение Gurulogic Multi-Variate Codec (GMVC®), поставляемое компанией Gurulogic Microsystems Oy. Кодек GMVC® способен кодировать различные типы данных с высокой степенью эффективности, формируя несколько различных потоков данных, которые содержат всю исходную входную информацию, а именно, входные данные (D1), обеспечивая их эффективное энтропийное кодирование. К примеру, в упомянутом выше фирменном кодеке GMVC® для кодирования данных изображений и видеоданных применяют различные способы, дающие различные потоки данных, в зависимости от содержимого входных данных (D1), как это описано в патентном документе GB 2503295 A («Кодер и соответствующий способ»), а также в документе GB 2505169 A («Декодер и соответствующий способ»). Поэтому предпочтительно, чтобы перед шифрованием, в результате которого получают кодированные и шифрованные данные (E2), для различных типов данных обеспечивалось эффективное сжатие с использованием различных энтропийных кодеров, оптимальных именно для этих конкретных типов данных, с учетом также разрядности и энтропии входных данных (D1). С этой целью, опционально, упомянутая схема обработки данных, например, реализованная с использованием одного или более упомянутых RISC-процессоров, в кодере 110 выполнена с возможностью кодирования и шифрования входных данных (D1) с целью формирования соответствующих кодированных и шифрованных данных (E2), последовательно и повторно, с ветвлением на множество параллельных последовательностей операций по кодированию и шифрованию блоков данных и/или пакетов данных с использованием соответствующих начальных значений; при этом RISC-процессор способен выполнять сравнительно простые операции конкатенации данных с очень высокой скоростью, и хорошо подходит для кодирования и декодирования данных, поступающих в потоковом формате, например, в реальном времени. Благодаря этому несколько одновременных процедур могут исполняться параллельно, что обеспечивает ускоренную обработку входных данных (D1), а также повышает степень криптографической защиты, поскольку в декодере 120 должен применяться такой же тип разветвления при декодировании кодированных и шифрованных данных (E2). Опционально, для реализации ветвления входные данные (D1) подразделяют на различные потоки блоков данных и/или пакетов данных, например, для каждого типа данных, содержащихся во входных данных (D1).

Предпочтительно, для различных потоков применяют различные алгоритмы кодирования и шифрования. Опционально, альтернативно или в дополнение, ветвление, из практических соображений, может быть реализовано в целях ускорения процедур кодирования и шифрования, даже когда входные данные (D1) содержат данные только одного типа. В этом отношении, опционально, кодированные и шифрованные данные (E2) включают информацию, указывающую на то, каким образом входные данные (D1) были подразделены для реализации ветвления, которую используют при последующем дешифровании и декодировании кодированных и шифрованных данных (E2).

Опционально, упомянутая схема обработки данных в кодере 110 выполнена с возможностью использования упомянутого по меньшей мере одного ключа многократно, в комбинации с упомянутыми начальными значениями для шифрования кодированных блоков данных и/или пакетов данных с целью их включения в кодированные и шифрованные данные (E2). В одном из примеров схема обработки данных в кодере 110, опционально, выполнена с возможностью добавления упомянутых начальных значений в начало упомянутого по меньшей мере одного ключа для шифрования кодированных блоков данных и/или пакетов данных. В другом примере схема обработки данных в кодере 110, опционально, выполнена с возможностью добавления упомянутых начальных значений в конец упомянутого по меньшей мере одного ключа для шифрования кодированных блоков данных и/или пакетов данных. Альтернативно и опционально, упомянутая схема обработки данных в кодере 110 выполнена с возможностью использования упомянутого по меньшей мере одного ключа для шифрования только упомянутого первого кодированного блока данных и/или пакета данных. В этом случае, опционально, упомянутая схема обработки данных в кодере 110 выполнена с возможностью использования только упомянутых начальных значений для шифрования последующих кодированных блоков данных и/или пакетов данных с целью их включения в кодированные и шифрованные данные (E2). Другими словами, упомянутый первый кодированный блок данных и/или пакет данных шифруют с использованием только упомянутого по меньшей мере одного ключа.

При этом, опционально, упомянутая схема обработки данных в кодере 110 выполнена с возможностью применения инициализационного вектора (IV) в комбинации с упомянутым по меньшей мере одним ключом для шифрования упомянутого первого кодированного блока данных и/или пакета данных.

Также, в одной из моделей технической реализации, которая может быть применена в одном из вариантов осуществления настоящего изобретения, в кодере 110 запрограммирована функциональность прерывания, которая обеспечивает возможность формирования начальных значений, а именно, вызова подпрограмм или функций для формирования начальных значений.

В одном из вариантов осуществления настоящего изобретения информацию, которую используют для формирования начальных значений, определяют в декодере 120 или предоставляют в декодер 120, что обеспечивает возможность выполнение обратной процедуры при последующем дешифровании. С этой целью, опционально, упомянутая схема обработки данных в кодере 110 выполнена с возможностью включения в кодированные и шифрованные данные (E2) информации, указывающей на по меньшей один алгоритм, использованный для формирования начальных значений с целью их использования при шифровании кодированных блоков данных и/или пакетов данных. Обеспечение различных начальных значений для различных блоков данных и/или пакетов данных повышает эффективность процедуры шифрования.

Начальные значения могут вычисляться на основе практически любого типа информации, присутствующей во входных данных (D1). Однако нужно понимать, что начальные значения, предпочтительно, не вычисляют на основе обработанных данных, представленных кодированными и шифрованными данными (E2), поскольку в таком случае интеграция процедур шифрования и кодирования не позволит получить начальных значений, формирование которых при последующем дешифровании в декодере 120 потребует интерпретации процедур шифрования и кодирования. Иными словами, начальные значения, предпочтительно, вычисляют на основе переменной текущей информации, формируемой процедурой кодирования, которая не может быть непосредственно получена исключительно на основе кодированных и шифрованных данных (E2). Опционально, начальные значения запрашиваются процедурой при помощи изменения параметров.

В одной из реализаций начальные значения вычисляют до выполнения процедуры кодирования, например, с использованием контрольной суммы или хэш-функции. Теоретически, это несколько упрощает соответствующую процедуру дешифрования, однако все равно обеспечивает преимущества по сравнению со способами, известными на существующем уровне техники. В одном из примеров контрольную сумму или хэш-функцию вычисляют по меньшей мере над частью заданного блока данных и/или пакета данных из входных данных (D1). Нужно понимать, что во избежание обратного инжиниринга не допускается получение начальных значений на основе двусторонних исходных значений. При этом также, во избежание использования вероятностных вычислений, не допускается формирование начальных значений на основе предсказания. Предпочтительно, начальные значения вычисляют с использованием односторонних алгоритмов хэширования.

В другой реализации начальные значения вычисляют до выполнения процедуры шифрования с использованием режима сцепления блоков шифрованного текста (Cipher-Block Chaining, CBC) в алгоритме шифрования по расширенному стандарту шифрования (Advanced Encryption Standard, AES). В таком случае шифрованный результат, то есть, шифрованный текст, для предыдущего блока данных и/или пакета данных комбинируют с шифруемым блоком данных и/или пакета данных, то есть, с открытым текстом. Это означает, что режим CBC алгоритма шифрования AES функционирует стандартным образом во всех отношениях, кроме того, что начальные значения, формируемые в процессе кодирования, внедряют в шифрованные выходные данные, а именно, в кодированные и шифрованные данные (E2). Процедура шифрования может также выполняться с использованием потокового шифрования, а именно, шифрования с симметричным ключом, при котором цифры открытого текста комбинируют с псевдослучайным потоком цифр шифра, а именно, «ключевым потоком». В таком случае блоки данных и/или пакеты данных заменяют на один или более потоков данных. В вариантах осуществления настоящего изобретения могут применяться несколько различных методов вычисления, использования и/или передачи начальных значений для подобного шифрования, основанного на начальных значениях. Используемый вариант может зависеть, например, от заданного сценария применения и/или назначения кодированных и шифрованных данных (E2), и/или от алгоритма шифрования, применяемого для шифрования.

В одном из альтернативных вариантов осуществления настоящего изобретения, опционально, из кодера 110 в декодер 120 передают информацию, которая по меньшей мере частично указывает на метод формирования начальных значений, если информация, используемая для формирования начальных значений, не может быть иным образом

определена в декодере 120 или предоставлена в декодер 120. В частности, это является предпочтительным в случае, когда начальные значения не связаны с входными данными (D1) или формируются случайным образом. Однако по соображениям безопасности желательно, чтобы начальные значения ни в коем случае не передавались непосредственно. При этом, опционально, кодер 110 выполнен с возможностью передачи кодированных и шифрованных данных (E2) на сервер данных и/или в хранилище данных (не показано на фиг. 1) для их хранения в базе данных (не показана на фиг. 1). Сервер и/или хранилище 108 данных, для возможности последующего декодирования кодированных данных (E2), сконфигурированы так, чтобы быть доступными для декодера 120, либо через сеть связи, или по прямому соединению, которые, предпочтительно, совместимыми с кодером 110.

В дополнение, опционально, кодер 110 выполнен с возможностью передачи по меньшей мере одного ключа и/или инициализационного вектора IV, и/или информации, указывающей на по меньшей мере один алгоритм, который был применен для формирования начальных значений, в сервер и/или хранилище данных для хранения в базе данных.

В некоторых из примеров декодер 120, опционально, выполнен с возможностью доступа к кодированным и шифрованным данным (E2), находящимся на сервере и/или в хранилище 108. В дополнение, опционально, декодер 120 выполнен с возможностью доступа по меньшей мере к одному ключу и/или инициализационному вектору IV, и/или информации, указывающей на по меньшей мере один алгоритм, который был применен для формирования начальных значений, из упомянутого сервера данных и/или другого сервера данных и/или хранилища данных. В альтернативных примерах кодер 110, опционально, выполнен с возможностью потоковой передачи кодированных и шифрованных данных (E2) в декодер 120, либо при помощи сети связи, либо по прямому соединению. При этом, следует отметить, что устройство, оснащенное аппаратным или программным кодером, может также осуществлять связь напрямую с другим устройством, также оснащенным аппаратным или программным кодером.

В еще одном из альтернативных примеров кодер 120, опционально, реализуют таким образом, чтобы он извлекал кодированные и шифрованные данные (E2) из машиночитаемого носителя, например, с жесткого диска или с твердотельного диска (Solid-State Drive, SSD).

Также, опционально, упомянутая схема обработки данных в кодере 110 выполнена с возможностью обеспечения передачи упомянутого по меньшей мере одного ключа из кодера 110 в декодер 120 для использования при последующем дешифровании и декодировании кодированных и шифрованных данных (E2). Опционально, упомянутый по меньшей мере один ключ предоставляют из кодера 100 в декодер 120 вручную, путем его передачи между соответствующими пользователями. Альтернативно, опционально, упомянутый по меньшей мере один ключ предоставляют из кодера 110 в декодер 120 при помощи шифрованного сообщения электронной почты, например, при помощи сообщения электронной почты, которое шифровано при помощи приложения Pretty Good Privacy (PGP), GNU Privacy Guard (GnuPG) или аналогичного средства. Также, альтернативно и опционально, упомянутый по меньшей мере один ключ передают из кодера 110 в декодер 120 по шифрованному соединению связи. Опционально, упомянутое шифрованное соединение связи реализуют при помощи протокола защищенных сокетов (SSL) / безопасности транспортного уровня (TLS). Декодер 120 включает схему обработки данных, которую используют для обработки кодированных и шифрованных данных (E2) с целью формирования соответствующих декодированных данных (D3).

Опционально, упомянутую схему обработки данных в декодере 120 реализуют с применением по меньшей мере одного процессора с архитектурой с ограниченным набором команд (RISC), который выполнен с возможностью исполнения программных инструкций, в соответствии с последующим более подробным описанием; при этом RISC-процессор способен выполнять сравнительно простые операции конкатенации данных с очень высокой скоростью, и хорошо подходит для кодирования и декодирования данных, поступающих в потоковом формате, например, в реальном времени.

Опционально, упомянутая схема обработки данных в декодере 120 выполнена с возможностью дешифрования и декодирования кодированных и шифрованных данных (E2), представленных в по меньшей мере одной из следующих форм: кодированные и шифрованные одномерные данные, кодированные и шифрованные многомерные данные, кодированные и шифрованные текстовые данные, кодированные и шифрованные двоичные данные, кодированные и шифрованные данные датчиков, кодированные и шифрованные аудиоданные, кодированные и шифрованные данные изображений, кодированные и шифрованные видеоданные, однако без ограничения перечисленным.

Как упоминалось выше, в упомянутую схему обработки данных в декодере 120, в процессе ее функционирования, предоставляют по меньшей мере один ключ, который используют при формировании декодированных данных (D3).

Схема обработки данных в декодере 120 выполнена с возможностью дешифрования первого кодированного и шифрованного блока данных и/или пакета данных из упомянутого множества шифрованных и кодированных блоков данных и/или пакетов данных с использованием упомянутого по меньшей мере одного ключа с целью формирования по меньшей мере первого кодированного блока данных и/или пакета данных. Схема обработки данных в декодере 120 в этом случае выполнена с возможностью декодирования первого кодированного блока данных и/или пакета данных, чтобы получить первый декодированный блок данных и/или пакет данных, который включают в декодированные данные (D3). В соответствии с предшествующим описанием, кодированные и шифрованные данные (E2), предпочтительно, включают информацию, которая указывает по меньшей мере один алгоритм, применяемый в кодере 110 для формирования начальных значений, используемых при шифровании кодированных блоков данных и/или пакетов данных. Схема обработки данных в декодере 120 выполнена с возможностью формирования, с помощью этой информации, первого начального значения для использования при дешифровании следующего кодированного и шифрованного блока данных и/или пакета данных из упомянутого множества шифрованных и кодированных блоков данных и/или пакетов данных с целью формирования следующего кодированного блока данных и/или пакета данных. Схема обработки данных в декодере 120 в этом случае выполнена с возможностью декодирования следующего кодированного блока данных и/или пакета данных, чтобы получить следующий декодированный блок данных и/или пакет данных, который включают в декодированные данные (D3).

Также упомянутая схема обработки данных в декодере 120 выполнена с возможностью формирования следующего начального значения для использования при дешифровании и декодировании последующего кодированного и шифрованного блока данных и/или пакета данных из упомянутого множества кодированных и шифрованных блоков данных и/или пакетов данных, и так последовательно и повторно до тех пор, пока упомянутое множество кодированных и шифрованных блоков данных

и/или пакетов данных, в виде кодированных и шифрованных данных (E2), не будет дешифровано и декодировано в декодированные данные (D3).

Таким образом, опционально, множество кодированных и шифрованных блоков данных и/или пакетов данных в виде кодированных и шифрованных данных (E2) дешифруют и декодируют по одному, поэтапно, в декодированные данные (D3).

Также, опционально, с целью формирования декодированных данных (D3), упомянутая схема обработки данных в декодере 120 выполнена с возможностью выполнения процедуры, обратной процедуре кодирования и шифрования, выполняемой схемой обработки данных в кодере 110. С этой целью, опционально, упомянутая схема обработки данных в декодере 120 выполнена с возможностью выполнения дополнительной процедуры декодирования над кодированными блоками данных и/или пакетами данных с применением по меньшей мере одного из следующего: декодирования энтропийным модификатором, дельта-декодирования, O-дельта-декодирования, диапазонного декодирования 1u или 8u, декодирования длин серий, скользящего декодирования длин серий и/или интерполирующего декодирования.

При этом, опционально, упомянутая схема обработки данных в декодере 120 выполнена с возможностью дешифрования кодированных и шифрованных данных (E2) с целью формирования соответствующих декодированных данных (D3), последовательно и повторно, с ветвлением на множество параллельных последовательностей операций по декодированию и дешифрованию шифрованных и кодированных блоков данных и/или пакетов данных с использованием соответствующих начальных значений. Опционально, для реализации ветвления, кодированные и шифрованные данные (E2) включают различные потоки блоков данных и/или пакетов данных, например, для каждого типа данных, содержащихся во входных данных (D1).

Опционально, упомянутая схема обработки данных в декодере 120 выполнена с возможностью использования упомянутого по меньшей мере одного ключа многократно, в комбинации с упомянутыми начальными значениями для дешифрования кодированных и шифрованных блоков данных и/или пакетов данных из кодированных и шифрованных данных (E2). В одном из примеров схема обработки данных в кодере 120, опционально, выполнена с возможностью добавления упомянутых начальных значений в начало упомянутого по меньшей мере одного ключа для дешифрования кодированных и шифрованных блоков данных и/или пакетов данных.

В другом примере схема обработки данных в кодере 120, опционально, выполнена с возможностью добавления упомянутых начальных значений в конец упомянутого по меньшей мере одного ключа для дешифрования кодированных и шифрованных блоков данных и/или пакетов данных.

Альтернативно и опционально, упомянутая схема обработки данных в декодере выполнена с возможностью применения упомянутого по меньшей мере одного ключа для дешифрования только упомянутого первого кодированного и шифрованного блока данных и/или пакета данных. В этом случае схема обработки данных в кодере 120, опционально, выполнена с возможностью применения только упомянутых начальных значений дешифрования последующего кодированного и шифрованного блока данных и/или пакета данных. Также, опционально, упомянутая схема обработки данных в декодере 120 выполнена с возможностью применения инициализационного вектора (IV) в комбинации с упомянутым по меньшей мере одним ключом для дешифрования упомянутого первого кодированного и шифрованного блока данных и/или пакета данных.

Также, опционально, во время кодирования, заданный блок данных и/или пакет

данных обрабатывают таким образом, что в результате получают начальное значение, которое используют при шифровании следующего за ним блока данных и/или пакета данных. То есть, начальное значение получают на основе содержимого заданного блока данных и/или пакета данных. Таким образом, при дешифровании и декодировании, чтобы была возможность корректно дешифровать и декодировать следующий кодированный и шифрованный блок данных и/или пакет данных, должен быть корректно дешифрован и декодирован заданный кодированный и шифрованный блок данных и/или пакет данных. Соответственно, формирование декодированных данных (D3) на основе кодированных и шифрованных данных (E2) требует дешифрования и декодирования множества кодированных и шифрованных блоков данных и/или пакетов данных по одному и поэтапно.

Описанная выше интеграция процедур кодирования и шифрования увеличивает объем вычислительных ресурсов, необходимых для взлома выполняемого таким образом шифрования. В результате время, необходимое для попытки взлома шифрования, делает задачу взлома еще более сложной.

С технической точки зрения более экономичным является кодирование, например, при помощи сжатия данных, множества блоков данных и/или пакетов данных из входных данных (D1) перед их шифрованием, поскольку в таком случае энтропия и размер кодированных и шифрованных данных (E2) будет меньшим, чем в том случае, если бы входные данные (D1) были шифрованы до сжатия. Алгоритм шифрования, как правило, создает максимальную энтропию в виде кодированных и шифрованных данных (E2), что с точки зрения математики означает наличие максимально возможного теоретически количества альтернативных вариантов для дешифрования кодированных и шифрованных данных (E2). Нужно понимать, что сжатие, опционально, применяют для входных данных (D1) в целом, а не к отдельным блокам данных и/или пакетам данных из их состава. В таком случае применяют механизм временного безопасного хранения начальных значений, сформированных при сжатии, то есть, кодировании, например, в виде шифрованных данных.

На фиг. 1 показан лишь один из примеров, который не ограничивает объем настоящего изобретения, заданный в приложенной формуле изобретения. Нужно понимать, что конкретная структура кодека 130 приведена исключительно в качестве примера и не должна рассматриваться как ограничивающая кодек 130 конкретным количеством, типами или схемой размещения кодеров и декодеров. Специалистам в данной области техники могут быть очевидны множество вариаций, альтернатив и изменений в вариантах осуществления настоящего изобретения.

Опционально, кодек 130 реализуют внутри одного устройства. Альтернативно и опционально, кодек 130, из практических соображений, реализуют с распределением по различным устройствам. Опционально, кодек 130 может быть реализован в качестве цифрового аппаратного обеспечения, разработанного по заказу например, при помощи одной или более заказных интегральных схем (application-specific integrated circuits, ASIC). Альтернативно или в дополнение, кодек может быть реализован в виде компьютерных программных инструкций, исполняемых на вычислительной аппаратуре.

Кодек 130 может быть реализован в форме по меньшей мере одного из следующего: кодек данных, аудиокодек, кодек изображений и/или видеокодек, но без ограничения перечисленным.

При этом кодек 130 может быть реализован для обеспечения защищенной связи между передатчиками и приемниками, и одновременно с возможностью значительной экономии полосы пропускания сети, необходимой для передачи данных, а также без

необходимости шифрованного соединения, такого как SSL/TLS, при передаче данных. В одном из примеров кодек 130 может быть реализован в системе, основанной на связи типа «запрос-ответ», например, протоколе передаче гипертекста (HyperText Transfer Protocol, HTTP), который используют для передачи данных в веб-браузерах и серверах всемирной сети Интернет (World Wide Web, www). Несмотря на ненулевую вероятность того, что данные, шифруемые существующими алгоритмами, могут быть вскрыты и дешифрованы методом прямого перебора в будущем, предполагается, что будущие алгоритмы шифрования, соответственно, будут формировать более надежные, по сравнению с современными алгоритмами, ключи шифрования, что обеспечит более надежное шифрование данных.

Помимо прямого перебора существуют и другие широко известные методы взлома, такие как атака по полному двудольному графу, атака на основе связанных ключей, атака с оракулом, атака увеличением длины сообщения и другие, однако эти методы по существу не дают результатов при взломе шифрования, выполняемого кодером 110.

Исключительно для иллюстрации ниже будет приведен технический пример процедуры шифрования, исполняемый в кодере 110. В данном примере представлена эффективная в большинстве случаев модель шифрования нешифрованного потока открытых данных при помощи симметричного алгоритма шифрования AES в режиме CBC и начального значения с расширенным ключом шифрования, которая соответствует следующим шагам:

1. Получение или формирование двух ключей шифрования, а именно Key1 и Key2.

2. Формирование байтов случайного криптографического инициализационного вектора (IV) для режима CBC алгоритма AES.

3. Шифрование байтов открытого текста (а именно, кодированных блоков данных и/или пакетов данных), с получением байтов шифрованного текста (а именно, кодированных и шифрованных блоков данных и/или пакетов данных) с использованием функции AES CBC, с использованием Key1 и IV, или Key1 и Salt.

4. Слияние байтов инициализационного вектора и байтов шифрованного текста.

5. Формирование байтов кода проверки подлинности сообщения (Message Authentication Code, MAC) при помощи функции HMAC с параметрами Key2 и Ciphertext.

6. Запись байтов MAC и шифрованного текста в поток данных, а именно, в кодированные и шифрованные данные (E2).

При этом псевдокод для описанного выше алгоритма имеет следующий вид:

```
Key1 = KeyStretch(GetKey())
Key2 = KeyStretch(GetKey())
IV = Random()
Ciphertext = IV + AES(Key1 + Salt, IV, Plaintext)
MAC = HMAC(Key2, Ciphertext)
DATA = MAC + Ciphertext
```

В рассмотренном выше примере два усиленных ключа были получены с помощью метода «растяжения ключа». Метод растяжения ключа реализуют, как правило, тысячи раз пропуская пароль для шифрования через односторонний алгоритм свертывания, то есть, алгоритм хэширования. Это дает достаточно изменений, позволяющих защитить пароль от атак, а именно, атак на основе связанных ключей.

После этого для режима CBC формируют соответствующие байты инициализационного вектора (IV). Байты инициализационного вектора затем скремблируют и внедряют в первый кодированный блок данных и/или пакет данных, подлежащий шифрованию. Первый кодированный блок данных и/или пакет данных

затем шифруют при помощи симметричного алгоритма шифрования AES в режиме CBC с использованием многократно растянутого ключа и байтов инициализационного вектора. Использование байтов инициализационного вектора для первого кодированного блока данных и/или пакета данных особенно предпочтительно в целях повышения степени защиты за счет шифрования, например, в случае, когда входные данные (D1) содержат большое количество избыточных данных. В результате взломщик не сможет дешифровать полную последовательность информации, а именно, получить входные данные (D1), если не будут вскрыты все кодированные и шифрованные блоки данных и/или пакеты данных в виде кодированных и шифрованных данных (E2), от начала до конца.

Затем последующие кодированные блоки данных и/или пакеты данных шифруют при помощи симметричного алгоритма шифрования AES в режиме CBC с использованием многократно растянутого ключа и начальных значений, сформированных в процедуре кодирования.

Наконец, в шифрованный текст, то есть в кодированные и шифрованные данные (E2), внедряют байты кода проверки подлинности сообщений (MAC). Это исключает возможность получения идентичных шифрованных текстов из-за случайного наличия избыточного открытого текста во входных данных (D1), а также исключает взлом данных, например, при помощи атаки с оракулом. Также, это гарантирует сохранение целостности кодированных и шифрованных данных (E2). Варианты осуществления настоящего изобретения, представленные на чертежах, были рассмотрены на примере режима сцепления блоков шифрованного текста (CBC), однако при этом нужно понимать, что варианты осуществления настоящего изобретения могут быть также реализованы при помощи поточного шифрования, при котором блоки данных и/или пакеты данных заменяют на один или более потоков данных.

Обратимся к фиг. 2, на которой представлена блок-схема алгоритма, иллюстрирующая шаги первого интегрированного способа кодирования и шифрования входных данных (D1), включающих множество блоков данных и/или пакетов данных, с целью формирования соответствующих кодированных и шифрованных данных (E2), в соответствии с одним из вариантов осуществления настоящего изобретения. Способ проиллюстрирован в виде набора шагов на блок-схеме алгоритма, последовательность шагов которого может быть реализована в виде аппаратного обеспечения, программного обеспечения или их комбинации, например, в соответствии с предшествующим описанием.

Исключительно в целях иллюстрации, способ будет рассмотрен ниже на примере кодера 110, показанного на фиг. 1.

На шаге 202 схема обработки данных в кодере 110 считывает или принимает первый блок данных и/или пакет данных из упомянутого множества блоков данных и/или пакетов данных.

Затем, на шаге 204, схема обработки данных в кодере 110 кодирует первый блок данных и/или пакет данных из упомянутого множества блока данных и/или пакета данных с целью формирования первого кодированного блока данных и/или пакета данных. Опционально, на шаге 204, схема обработки данных в кодере 110 обрабатывает первый блок данных и/или пакет данных и формирует первое начальное значение для использования при шифровании следующего блока данных и/или пакета данных из упомянутого множества блоков данных и/или пакетов данных. Это начальное значение может в дальнейшем описании называться также «солью».

Затем, на шаге 206, схема обработки данных в кодере 110 шифрует первый блок

данных и/или пакет данных с использованием упомянутого по меньшей мере одного ключа, чтобы получить первый кодированный и шифрованный блок данных и/или пакет данных, который включают в кодированные и шифрованные данные (E2). Опционально, в схему обработки данных в кодере 110 предоставляют по меньшей мере один ключ, который используют при формировании кодированных и шифрованных данных (E2). Альтернативно и опционально, схема обработки данных в кодере 110 формирует упомянутый по меньшей мере один ключ с использованием подходящего алгоритма формирования ключа.

В дополнение, опционально, схема обработки данных в кодере 110 применяет инициализационный вектор (IV) в комбинации с упомянутым по меньшей мере одним ключом для шифрования упомянутого первого кодированного блока данных и/или пакета данных на шаге 206.

Затем, на шаге 208, схема обработки данных в кодере 110 записывает или передает первый кодированный и шифрованный блок данных и/или пакет данных в кодированные и шифрованные данные (E2).

На шаге 210 схема обработки данных в кодере 110 считывает или принимает следующий блок данных и/или пакет данных из упомянутого множества блоков данных и/или пакетов данных.

Затем, на шаге 212, схема обработки данных в кодере 110 кодирует следующий блок данных и/или пакет данных из упомянутого множества блока данных и/или пакета данных с целью формирования следующего кодированного блока данных и/или пакета данных.

Опционально, на шаге 212, схема обработки данных в кодере 110 обрабатывает упомянутый следующий блок данных и/или пакет данных и формирует следующее начальное значение для использования при шифровании последующего блока данных и/или пакета данных из упомянутого множества блоков данных и/или пакетов данных.

Затем, на шаге 214, схема обработки данных в кодере 110 шифрует следующий кодированный блок данных и/или пакет данных с использованием упомянутого первого начального значения, сформированного на шаге 204, в комбинации с упомянутым по меньшей мере одним ключом, чтобы получить следующий кодированный и шифрованный блок данных и/или пакет данных, который включают в кодированные и шифрованные данные (E2). С этой целью может выполняться слияние первого начального значения и упомянутого по меньшей мере одного ключа множеством различных путей. В качестве примера, упомянутый по меньшей мере один ключ «солят», добавляя в его начало или конец упомянутое первое начальное значение. Затем, на шаге 216, схема обработки данных в кодере 110 записывает или передает следующий кодированный и шифрованный блок данных и/или пакет данных в кодированные и шифрованные данные (E2).

Аналогично, на шаге 218 схема обработки данных в кодере 110 считывает или принимает последующий блок данных и/или пакет данных из упомянутого множества блоков данных и/или пакетов данных.

Затем, на шаге 220, схема обработки данных в кодере 110 кодирует последующий блок данных и/или пакет данных из упомянутого множества блока данных и/или пакета данных с целью формирования последующего кодированного блока данных и/или пакета данных.

Опционально, на шаге 220, схема обработки данных в кодере 110 обрабатывает упомянутый последующий блок данных и/или пакет данных и формирует последующее начальное значение (не показано на фиг. 2) для использования при шифровании

последующего блока данных и/или пакета данных из упомянутого множества блоков данных и/или пакетов данных.

Затем, на шаге 222, схема обработки данных в кодере 110 шифрует последующий кодированный блок данных и/или пакет данных с использованием упомянутого следующего начального значения, сформированного на шаге 212, в комбинации с упомянутым по меньшей мере одним ключом, чтобы получить последующий кодированный и шифрованный блок данных и/или пакет данных, который включают в кодированные и шифрованные данные (E2). С этой целью может выполняться слияние упомянутого следующего начального значения и упомянутого по меньшей мере одного ключа, при помощи добавления следующего начального значения в начало или в конец упомянутого по меньшей мере одного ключа.

Затем, на шаге 224, схема обработки данных в кодере 110 записывает или передает последующий кодированный и шифрованный блок данных и/или пакет данных в кодированные и шифрованные данные (E2).

Шаги 218-224 выполняют, и так последовательно и повторно до тех пор, пока все множество блоков данных и/или пакетов данных из входных данных (D1) не будет кодировано и шифровано в кодированные и шифрованные данные (E2).

Шаги 202-224 являются исключительно примерами, и соответственно, в пределах объема настоящего изобретения могут быть предложены также различные альтернативы, в которые добавлены один или более шагов, один или более шагов опущены, или один или более шагов выполняют в другой последовательности. В одном из альтернативных вариантов осуществления настоящего изобретения кодирование упомянутого множества блоков данных и/или пакетов данных выполняют до начала шифрования соответствующих им кодированных блоков данных и/или пакетов данных. Другими словами, шаги 202, 204, 210, 212, 218 и 220, а именно, шаги, относящиеся к чтению и кодированию блоков данных и/или пакетов данных из входных данных (D1), выполняют перед шагами 206, 208, 214, 216, 222 и 224, а именно, перед шагами, относящимися к шифрованию кодированных блоков данных и/или пакетов данных и записи в кодированные и шифрованные данные (E2).

В соответствии с иллюстрацией фиг. 2, шаги 206, 214 и 222, опционально, выполняют при помощи режима CBC симметричного алгоритма шифрования AES. Шаг 206, опционально, выполняют с использованием сформированного случайным образом инициализационного вектора, который комбинируют с упомянутым по меньшей мере одним ключом. Шаги 214 и 222 выполняют с использованием соответствующих им начальных значений («соли»), сформированных на шагах 204 и 212 соответственно. В одном из примеров в качестве «соли» для блока данных и/или пакета данных, следующего за заданным блоком данных и/или пакетом данных, используют соответствующее связанное с ним базовое 64-битное значение.

Нужно понимать, что способ, проиллюстрированный на фиг. 2, может быть реализован с использованием других алгоритмов шифрования, независимо от того, применяют ли инициализационный вектор для шифрования первого кодированного блока данных и/или пакета данных, и независимо от того, применяют ли режим CBC. В вариантах осуществления настоящего изобретения предложен компьютерный программный продукт, включающий машиночитаемый носитель, на котором хранят машиночитаемые инструкции, при этом машиночитаемые инструкции могут быть исполнены на компьютеризованном устройстве, что обеспечивает исполнение первого интегрированного способа, описанного в связи с фиг. 2. Машиночитаемые инструкции, опционально, могут быть загружены в компьютеризованное устройство из магазина

программных приложений, например, "App store".

Фиг. 3 представляет собой эскизную блок-схему алгоритма, которая иллюстрирует шаги второго интегрированного способа дешифрования и декодирования кодированных и шифрованных данных (E2), включающих множество кодированных и шифрованных блоков данных и/или пакетов данных, с целью формирования соответствующих декодированных данных (D3), в соответствии с одним из вариантов осуществления настоящего изобретения; Способ проиллюстрирован в виде набора шагов на блок-схеме алгоритма, последовательность шагов которого может быть реализована в виде аппаратного обеспечения, программного обеспечения или их комбинации.

Исключительно в целях иллюстрации, способы будет рассмотрен ниже на примере декодера 120, показанного на фиг. 1.

На шаге 302 схема обработки данных в декодере 120 считывает или принимает первый кодированный и шифрованный блок данных и/или пакет данных из упомянутого множества кодированных и шифрованных блоков данных и/или пакетов данных. Затем, на шаге 304, схема обработки данных в декодере 120 дешифрует первый кодированный и шифрованный блок данных и/или пакет данных с использованием по меньшей мере одного ключа, чтобы получить первый кодированный блок данных и/или пакет данных.

Опционально, в упомянутую схему обработки данных в декодере 120 предоставляют упомянутый по меньшей мере один ключ, который используют при формировании декодированных данных (D3).

В дополнение, опционально, схема обработки данных в декодере 120 применяет инициализационный вектор (IV) в комбинации с упомянутым по меньшей мере одним ключом для дешифрования упомянутого первого кодированного и шифрованного блока данных и/или пакета данных на шаге 304.

Затем, на шаге 306, схема обработки данных в декодере 120 декодирует первый кодированный блок данных и/или пакет данных, чтобы получить первый декодированный блок данных и/или пакет данных, который включают в декодированные данные (D3).

Опционально, на шаге 306, схема обработки данных в декодере 120 обрабатывает первый декодированный блок данных и/или пакет данных и формирует первое начальное значение для использования при дешифровании следующего кодированного и шифрованного блока данных и/или пакета данных из упомянутого множества кодированных и шифрованных блоков данных и/или пакетов данных.

Затем, на шаге 308, схема обработки данных в декодере 120 записывает или передает первый декодированный блок данных и/или пакет данных в декодированные данные (D3).

На шаге 310 схема обработки данных в декодере 120 считывает или принимает следующий кодированный и шифрованный блок данных и/или пакет данных из упомянутого множества кодированных и шифрованных блоков данных и/или пакетов данных.

Затем, на шаге 312, схема обработки данных в декодере 120 дешифрует упомянутый следующий кодированный и шифрованный блок данных и/или пакет данных с использованием первого начального значения, сформированного на шаге 306, в комбинации с упомянутым по меньшей мере одним ключом, чтобы получить следующий кодированный блок данных и/или пакет данных.

С этой целью может выполняться слияние первого начального значения и упомянутого по меньшей мере одного ключа множеством различных образов. В качестве примера, упомянутый по меньшей мере один ключ «солят», добавляя в его начало или

конец упомянутое первое начальное значение.

Затем, на шаге 314, схема обработки данных в декодере 120 декодирует следующий кодированный блок данных и/или пакет данных, чтобы получить следующий декодированный блок данных и/или пакет данных, который включают в декодированные данные (D3).

Опционально, на шаге 314, схема обработки данных в декодере 120 обрабатывает упомянутый следующий декодированный блок данных и/или пакет данных и формирует следующее начальное значение для использования при дешифровании последующего кодированного и шифрованного блока данных и/или пакета данных из упомянутого множества кодированных и шифрованных блоков данных и/или пакетов данных.

Затем, на шаге 316, схема обработки данных в декодере 120 записывает или передает упомянутый следующий декодированный блок данных и/или пакет данных в декодированные данные (D3). Аналогично, на шаге 318 схема обработки данных в декодере 120 считывает или принимает упомянутый последующий кодированный и шифрованный блок данных и/или пакет данных из упомянутого множества кодированных и шифрованных блоков данных и/или пакетов данных. Затем, на шаге 320, схема обработки данных в декодере 120 дешифрует упомянутый последующий кодированный и шифрованный блок данных и/или пакет данных с использованием упомянутого следующего начального значения, сформированного на шаге 314, в комбинации с упомянутым по меньшей мере одним ключом, чтобы получить последующий кодированный блок данных и/или пакет данных. С этой целью может выполняться слияние упомянутого следующего начального значения и упомянутого по меньшей мере одного ключа, при помощи добавления следующего начального значения в начало или в конец упомянутого по меньшей мере одного ключа.

Затем, на шаге 322, схема обработки данных в декодере 120 декодирует упомянутый последующий кодированный блок данных и/или пакет данных, чтобы получить последующий декодированный блок данных и/или пакет данных, который включают в декодированные данные (D3).

Опционально, на шаге 322, схема обработки данных в декодере 120 обрабатывает упомянутый последующий декодированный блок данных и/или пакет данных и формирует последующее начальное значение (не показано на фиг. 3) для использования при дешифровании последующего кодированного и шифрованного блока данных и/или пакета данных из упомянутого множества кодированных и шифрованных блоков данных и/или пакетов данных.

Затем, на шаге 324, схема обработки данных в декодере 120 записывает или передает упомянутый последующий декодированный блок данных и/или пакет данных в декодированные данные (D3).

Шаги 318-324 выполняют, и так последовательно и повторно до тех пор, пока все множество кодированных и шифрованных блоков данных и/или пакетов данных из кодированных и шифрованных данных (E2) не будет дешифровано и декодировано в декодированные данные (D3).

Шаги 302-324 являются исключительно примерами, и соответственно, в пределах объема настоящего изобретения могут быть предложены также различные альтернативы, в которых добавлены один или более шагов, один или более шагов опущены, или один или более шагов выполняют в другой последовательности.

В соответствии с иллюстрацией фиг. 3, шаги 304, 312 и 320, опционально, выполняют при помощи режима CBC симметричного алгоритма шифрования AES. Шаг 304, опционально, выполняют с использованием сформированного случайным образом

инициализационного вектора, который комбинируют с упомянутым по меньшей мере одним ключом. Шаги 312 и 320 выполняют с использованием соответствующих им значений «соли», сформированных на шагах 306 и 314 соответственно. В одном из примеров в качестве «соли» для кодированного и шифрованного блока данных и/или пакета данных, следующего за заданным декодированным блоком данных и/или пакетом данных, используют соответствующее связанное с ним 64-битное значение.

Нужно понимать, что способ, проиллюстрированный на фиг. 3, может быть реализован с использованием других алгоритмов шифрования, независимо от того, применяют ли инициализационный вектор для дешифрования первого кодированного блока данных и/или пакета данных, и независимо от того, применяют ли режим CBC.

В вариантах осуществления настоящего изобретения предложен компьютерный программный продукт, включающий машиночитаемый носитель, на котором хранят машиночитаемые инструкции, при этом машиночитаемые инструкции могут быть исполнены на компьютеризованном устройстве, что обеспечивает исполнение второго интегрированного способа, описанного в связи с фиг. 3. Машиночитаемые инструкции, опционально, могут быть загружены в компьютеризованное устройство из магазина программных приложений, например, "App store".

Фиг. 4 представляет собой эскизную блок-схему алгоритма, которая иллюстрирует шаги третьего интегрированного способа кодирования и шифрования входных данных (D1), включающих множество блоков данных и/или пакетов данных, с целью формирования соответствующих кодированных и шифрованных данных (E2), в соответствии с еще одним из вариантов осуществления настоящего изобретения. Если не указано обратное, описание шагов первого интегрированного способа, проиллюстрированного на фиг. 2, остается верным, с соответствующими изменениями, и для шагов третьего интегрированного способа, проиллюстрированного на фиг. 4. А именно, описание шагов 202, 204, 206, 208, 210, 212, 216, 218, 220 и 224 остается верным, с соответствующими изменениями, для шагов 402, 404, 406, 408, 410, 412, 416, 418, 420 и 424 соответственно.

Шаг 414 отличается от шага 214 тем, что для шифрования упомянутого следующего блока данных и/или пакета данных используют только первое начальное значение, то есть, без упомянутого по меньшей мере одного ключа. Аналогично, шаг 422 отличается от шага 222 тем, что для шифрования упомянутого последующего кодированного блока данных и/или пакета данных используют только упомянутое следующее начальное значение. Альтернативно и опционально, схема обработки данных в кодере 110 выполнена с возможностью использования, на шаге 406, упомянутого по меньшей мере одного ключа для шифрования только упомянутого первого кодированного блока данных и/или пакета данных.

В вариантах осуществления настоящего изобретения предложен компьютерный программный продукт, включающий машиночитаемый носитель, на котором хранят машиночитаемые инструкции, при этом машиночитаемые инструкции могут быть исполнены на компьютеризованном устройстве, что обеспечивает исполнение третьего интегрированного способа, описанного в связи с фиг. 4. Машиночитаемые инструкции, опционально, могут быть загружены в компьютеризованное устройство из магазина программных приложений, например, "App store".

Фиг. 5 представляет собой эскизную блок-схему алгоритма, которая иллюстрирует шаги четвертого интегрированного способа дешифрования и декодирования кодированных и шифрованных данных (E2), включающих множество кодированных и шифрованных блоков данных и/или пакетов данных, с целью формирования

соответствующих декодированных данных (D3), в соответствии с еще одним из вариантов осуществления настоящего изобретения. Если не указано обратное, описание шагов первого интегрированного способа, проиллюстрированного на фиг. 3, остается верным, с соответствующими изменениями, и для шагов четвертого интегрированного способа, проиллюстрированного на фиг. 5. А именно, описание шагов 302, 304, 306, 308, 310, 314, 316, 318, 322 и 324 остается верным, с соответствующими изменениями, для шагов 502, 504, 506, 508, 510, 514, 516, 518, 522 и 524, соответственно.

Шаг 512 отличается от шага 312 тем, что для дешифрования упомянутого следующего кодированного и шифрованного блока данных и/или пакета данных используют только первое начальное значение, то есть, без упомянутого по меньшей мере одного ключа. Аналогично, шаг 520 отличается от шага 320 тем, что для дешифрования упомянутого последующего кодированного и шифрованного блока данных и/или пакета данных используют только упомянутое следующее начальное значение. То есть, в четвертом интегрированном способе схема обработки данных в кодере 120 выполнена с возможностью использования, на шаге 504, упомянутого по меньшей мере одного ключа для дешифрования только упомянутого первого кодированного и шифрованного блока данных и/или пакета данных.

В вариантах осуществления настоящего изобретения предложен компьютерный программный продукт, включающий машиночитаемый носитель, на котором хранят машиночитаемые инструкции, при этом машиночитаемые инструкции могут быть исполнены на компьютеризованном устройстве, что обеспечивает исполнение четвертого интегрированного способа, описанного в связи с фиг. 5. Машиночитаемые инструкции, опционально, могут быть загружены в компьютеризованное устройство из магазина программных приложений, например, "App store".

Описанные выше первый и третий интегрированные способы подходят для реализации в кодере или в устройстве предварительной обработки данных, связанном с кодером. К примеру, описанные выше первый и третий интегрированные способы подходят для применения в комбинации с известными на существующем уровне техники способами шифрования и могут служить дополнением к ним. Аналогично, описанные выше второй и четвертый интегрированные способы подходят для реализации в декодере или в устройстве предварительной обработки данных, связанном с декодером. К примеру, описанный выше второй и четвертый интегрированные способы подходят для применения в комбинации с известными на существующем уровне техники способами дешифрования и могут служить дополнением к ним. Описанные выше способы могут быть реализованы в виде программного обеспечения и/или при помощи жестко запрограммированной логики, например, заказных интегральных схем (ASIC). Общеизвестно, что многие системы имеют специальные микросхемы для шифрования, например, эффективно выполняющие шифрование по современному стандарту AES, потребляя при этом меньше электроэнергии, чем в случае чисто программной реализации. Описанные выше способы позволяют достичь значительного сокращения потребляемой мощности и электроэнергии, по сравнению с существующими решениями, в которых для предотвращения несанкционированного доступа к информации, например, против шпионского программного обеспечения, применяют шифрование соответствующей стойкости.

Варианты осуществления настоящего изобретения обеспечивают значительное усиление защиты по сравнению с защитой, обеспечиваемой при помощи способов, известных на существующем уровне техники, в которых применяют соответствующие алгоритмы шифрования. Способы, предложенные в вариантах осуществления

настоящего изобретения, могут быть реализованы с использованием любых подходящих методов кодирования, независимо от применяемого алгоритма шифрования. Благодаря этому, описанные выше способы не меняют поведения применяемых алгоритмов шифрования, и это означает, что эффективность защиты, обеспечиваемой встроенными алгоритмами шифрования, не будет снижена. Следовательно, способы, предложенные в вариантах осуществления настоящего изобретения, позволяют еще более повысить эффективность существующих алгоритмов сжатия данных и шифрования за счет их интеграции друг с другом и взаимосвязанного функционирования. Способы, предложенные в вариантах осуществления настоящего изобретения, могут эффективно применяться, например, в медицинских или военных целях для защиты ценной, конфиденциальной или секретной информации.

При этом предложенные способы могут быть реализованы в комбинации с распространенными общеизвестными программными приложениями для сжатия данных, как с открытым кодом, так и проприетарными, например, 7-Zip, Win-Zip и др.

При этом также, комбинирование процедур шифрования и кодирования представляет собой эффективную модель для многопроцессорной обработки данных или для параллельного запуска нескольких процедур. Интеграция процедур друг с другом позволяет реализовать оптимальную структуру обработки данных для каждого заданного центрального процессорного блока (CPU) и заданного графического процессора (GPU), в соответствии с доступной вычислительной мощностью. То есть, описанные выше способы позволяют реализовать эффективное разделение на потоки для процедуры шифрования, встроенной в процедуру кодирования, при этом блоки данных и/или пакеты данных из входных данных (D1) могут быть представлены в оптимизированном формате, оптимальном для CPU и GPU системы и/или платформы, на которой выполняют процедуру кодирования. Описанное выше ветвление при шифровании и дешифровании позволяет применять методы параллельной обработки данных, например, распределяя обработку отдельных данных по разным ветвям алгоритма. Таким образом, становится возможным применение такого вычислительного оборудования, как цифровые матричные процессоры (digital array processors, DAP) для высокоскоростного шифрования и соответствующего дешифрования данных.

Описанные выше способы позволяют применять очень быстрые, но вместе с тем в высокой степени эффективные алгоритмы шифрования. В этом отношении предложенные способы обеспечивают эффективное применение алгоритмов шифрования без вмешательства в их внутреннее функционирование. Примеры алгоритмов шифрования, которые подходят для реализации в сочетании с упомянутыми выше способами, включают, без ограничения перечисленным, AES, Twofish, Blowfish, стандарт шифрования данных (Data Encryption Standard, DES), Triple DES (3-DES), международный алгоритм шифрования данных (International Data Encryption Algorithm, IDEA), MARS, Rivest Cipher 6 (RC6), Camellia, CAST-128, Skipjack, расширенный миниатюрный алгоритм шифрования данных (extended Tiny Encryption Algorithm, XTEA).

При этом дополнительное преимущество от встраивания процедуры шифрования в процедуру кодирования заключается в том, что полученные таким образом кодированные и шифрованные данные (E2) не обязательно нужно передавать в другие сети с использованием защищенного безопасного сетевого соединения, например, с применением тоннеля виртуальной частной сети (Virtual Private Network, VPN), протоколов безопасной оболочки или протоколов SSL/TLS. Следовательно, рассмотренные выше способы являются более выгодной моделью передачи текстовых, двоичных, аудиоданных, данных изображений, видеоданных и данных других типов,

например, по публичным сетям Интернет или в веб-сервисах и облачных сервисах.

Варианты осуществления настоящего изобретения допускают применение в широком диапазоне систем и устройств, таких, например, как смартфоны, персональные компьютеры, аудиовизуальные устройства, фото- и видеокамеры, сети связи, устройства хранения данных, системы наблюдения, системы видеоконференцсвязи, медицинские устройства, сейсмографические устройства, разведывательные устройства, бортовые регистраторы типа «черный ящик», цифровые музыкальные инструменты, в которых применяют методы сэмплирования, без ограничения перечисленным. В пределах сущности и объема, заданных приложенной формулой изобретения, возможны модификации вариантов осуществления настоящего изобретения, приведенных в предшествующем описании. Такие выражения, как «включающий», «содержащий», «охватывающий», «состоящий из», «имеющий», «представляющий собой», которые использованы в описании и в формуле настоящего изобретения, должны пониматься как неисключающие, то есть допускающие также присутствие объектов, компонентов или элементов, явно не упомянутых. Использование единственного числа может также пониматься как относящееся к множественному числу. Числовые обозначения на приложенных чертежах, приведенные в скобках, имеют целью упрощение понимания пунктов формулы изобретения и не должны пониматься как ограничивающие изобретение, заявленное в этих пунктах формулы изобретения.

(57) Формула изобретения

1. Кодер (110) для кодирования и шифрования входных данных (D1), включающих множество блоков данных, или пакетов данных, или потоков данных, при этом кодер (110) включает схему обработки данных для обработки входных данных (D1) для формирования соответствующих кодированных и шифрованных данных (E2), при этом упомянутая схема обработки данных объединяет процессы кодирования и шифрования для формирования кодированных и шифрованных данных (E2), причем:

(i) упомянутая схема обработки данных выполнена с возможностью кодирования по меньшей мере первого блока данных, или пакета данных, или потока данных из упомянутого множества блоков данных, или пакетов данных, или потоков данных для формирования первого кодированного блока данных, или пакета данных, или потока данных и шифрования упомянутого по меньшей мере первого кодированного блока данных, или пакета данных, или потока данных с использованием по меньшей мере одного ключа, чтобы получить первый кодированный и шифрованный блок данных, или пакет данных, или поток данных для включения в кодированные и шифрованные данные (E2);

(ii) упомянутая схема обработки данных выполнена с возможностью формирования первого начального значения для использования при шифровании следующего кодированного блока данных, или пакета данных, или потока данных, чтобы получить следующий кодированный и шифрованный блок данных, или пакет данных, или поток данных для включения в кодированные и шифрованные данные (E2);

(iii) упомянутая схема обработки данных выполнена с возможностью формирования следующего начального значения для использования при шифровании последующего кодированного блока данных, или пакета данных, или потока данных, и так последовательно и повторно до тех пор, пока упомянутое множество блоков данных, или пакетов данных, или потоков данных не будет шифровано и кодировано в виде кодированных и шифрованных данных (E2),

при этом для данного кодируемого и шифруемого блока данных, или пакета данных,

или потока данных начальное значение формируют на основании предшествующего ему блока данных, или пакета данных, или потока данных.

2. Кодер (110) по п. 1, отличающийся тем, что в упомянутую схему обработки данных, в процессе ее функционирования, предоставляют по меньшей мере один ключ, который используют при формировании кодированных и шифрованных данных (E2).

3. Кодер (110) по п. 1 или 2, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью многократного использования упомянутого по меньшей мере одного ключа в комбинации с упомянутыми начальными значениями для шифрования кодированных блоков данных, или пакетов данных, или потоков данных для их включения в кодированные и шифрованные данные (E2).

4. Кодер (110) по любому из пп. 1-3, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью применения упомянутого по меньшей мере одного ключа для шифрования только упомянутого первого кодированного блока данных, или пакета данных, или потока данных.

5. Кодер (110) по любому из пп. 1-4, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью применения упомянутого по меньшей мере одного ключа для шифрования только упомянутого первого кодированного блока данных, или пакета данных, или потока данных.

6. Кодер (110) по любому из пп. 1-5, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью включения в кодированные и шифрованные данные (E2) информации, указывающей по меньшей мере один алгоритм, примененный для формирования начальных значений для использования при шифровании кодированных блоков данных, или пакетов данных, или потоков данных.

7. Кодер (110) по любому из пп. 1-6, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью кодирования и шифрования входных данных (D1) для формирования соответствующих кодированных и шифрованных данных (E2), последовательно и повторно, с ветвлением на множество параллельных последовательностей операций по кодированию и шифрованию блоков данных, или пакетов данных, или потоков данных с использованием соответствующих начальных значений.

8. Кодер (110) по любому из пп. 1-7, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью кодирования и шифрования входных данных (D1), представленных по меньшей мере в одной из следующих форм: одномерные данные, многомерные данные, текстовые данные, двоичные данные, данные датчиков, аудиоданные, данные изображений, видеоданные.

9. Кодер (110) по любому из пп. 1-8, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью обеспечения передачи упомянутого по меньшей мере одного ключа из кодера (110) для использования при последующем дешифровании и декодировании кодированных и шифрованных данных (E2), вручную или при помощи шифрованного сообщения электронной почты, или по шифрованному соединению связи.

10. Кодер (110) по п. 9, отличающийся тем, что упомянутое шифрованное соединение связи реализуют при помощи протокола защищенных сокетов (SSL) / безопасности транспортного уровня (TLS).

11. Кодер (110) по любому из пп. 1-10, отличающийся тем, что упомянутую схему обработки данных реализуют с применением по меньшей мере одного процессора с архитектурой с ограниченным набором команд (RISC), который выполнен с возможностью исполнения программных инструкций.

12. Способ кодирования и шифрования входных данных (D1), включающих множество блоков данных, или пакетов данных, или потоков данных, при помощи кодера (110), который включает схему обработки данных для обработки входных данных (D1) для формирования соответствующих кодированных и шифрованных данных (E2), при этом упомянутая схема обработки данных объединяет процессы кодирования и шифрования для формирования кодированных и шифрованных данных (E2), и способ включает:

(i) кодирование первого блока данных, или пакета данных, или потока данных из упомянутого множества блока данных, или пакета данных, или потока данных для формирования по меньшей мере первого кодированного блока данных, или пакета данных, или потока данных;

(ii) шифрование упомянутого по меньшей мере первого кодированного блока данных, или пакета данных, или потока данных с использованием по меньшей мере одного ключа, чтобы получить первый шифрованный и кодированный блок данных, или пакет данных, или поток данных для включения в кодированные и шифрованные данные (E2);

(iii) формирование первого начального значения для использования при шифровании следующего кодированного блока данных, или пакета данных, или потока данных, чтобы получить следующий кодированный и шифрованный блок данных, или пакет данных, или поток данных для включения в кодированные и шифрованные данные (E2); и

(iv) формирование следующего начального значения для использования при шифровании последующего кодированного блока данных, или пакета данных, или потока данных, и так последовательно и повторно до тех пор, пока упомянутое множество блоков данных, или пакетов данных, или потоков данных не будет шифровано и кодировано в виде кодированных и шифрованных данных (E2),

при этом для данного кодируемого и шифруемого блока данных, или пакета данных, или потока данных начальное значение формируют на основании предшествующего ему блока данных, или пакета данных, или потока данных.

13. Способ по п. 12, включающий предоставление, в упомянутую схему обработки данных, по меньшей мере одного ключа, который используют при формировании кодированных и шифрованных данных (E2).

14. Способ по п. 12 или 13, включающий многократное использование упомянутой схемой обработки данных упомянутого по меньшей мере одного ключа в комбинации с упомянутыми начальными значениями для шифрования кодированных блоков данных, или пакетов данных, или потоков данных для их включения в кодированные и шифрованные данные (E2).

15. Способ по любому из пп. 12-14, включающий применение упомянутой схемой обработки данных упомянутого по меньшей мере одного ключа для шифрования только упомянутого первого кодированного блока данных, или пакета данных, или потока данных.

16. Способ по любому из пп. 12-15, включающий применение, упомянутой схемой обработки данных, инициализационного вектора (IV) в комбинации с упомянутым по меньшей мере одним ключом при шифровании упомянутого первого кодированного блока данных, или пакета данных, или потока данных.

17. Способ по любому из пп. 12-16, включающий выполнение, упомянутой схемой обработки данных, включения в кодированные и шифрованные данные (E2) информации, указывающей по меньшей мере один алгоритм, примененный для формирования начальных значений для использования при шифровании кодированных

блоков данных, или пакетов данных, или потоков данных.

18. Способ по любому из пп. 12-17, включающий выполнение, упомянутой схемой обработки данных, кодирования и шифрования входных данных (D1) для формирования соответствующих кодированных и шифрованных данных (E2), последовательно и повторно, с ветвлением на множество параллельных последовательностей операций по кодированию и шифрованию блоков данных, или пакетов данных, или потоков данных с использованием соответствующих начальных значений.

19. Способ по любому из пп. 12-18, включающий выполнение, упомянутой схемой обработки данных, кодирования и шифрования входных данных (D1), представленных по меньшей мере в одной из следующих форм: одномерные данные, многомерные данные, текстовые данные, двоичные данные, данные датчиков, аудиоданные, данные изображений, видеоданные.

20. Способ по любому из пп. 12-19, включающий выполнение, упомянутой схемой обработки данных, передачи упомянутого по меньшей мере одного ключа из кодера (110) для использования при последующем дешифровании и декодировании кодированных и шифрованных данных (E2), вручную или при помощи шифрованного сообщения электронной почты, или по шифрованному соединению связи.

21. Способ по п. 20, включающий реализацию упомянутого соединения связи при помощи протокола защищенных сокетов (SSL) / безопасности транспортного уровня (TLS).

22. Способ по любому из пп. 12-21, включающий реализацию упомянутой схемы обработки данных с применением по меньшей мере одного процессора с архитектурой с ограниченным набором команд (RISC), который выполнен с возможностью исполнения программных инструкций.

23. Декодер (120) для дешифрования и декодирования кодированных и шифрованных данных (E2), включающих множество кодированных и шифрованных блоков данных, или пакетов данных, или потоков данных, при этом декодер (120) включает схему обработки данных для обработки кодированных и шифрованных данных (E2) для формирования соответствующих декодированных данных (D3), и в декодер (120), при его функционировании, предоставляют по меньшей мере один ключ, который используют при формировании декодированных данных (D3), при этом упомянутая схема обработки данных объединяет процессы декодирования и дешифрования для формирования декодированных данных (D3), причем:

(i) упомянутая схема обработки данных выполнена с возможностью дешифрования по меньшей мере первого кодированного и шифрованного блока данных, или пакета данных, или потока данных из упомянутого множества кодированных и шифрованных блоков данных, или пакетов данных, или потоков данных с использованием упомянутого по меньшей мере одного ключа, чтобы получить по меньшей мере первый кодированный блок данных, или пакет данных, или поток данных, и декодирования упомянутого по меньшей мере первого кодированного блока данных, или пакета данных, или потока данных, чтобы получить первый декодированный блок данных, или пакет данных, или поток данных для включения в декодированные данные (D3);

(ii) упомянутая схема обработки данных выполнена с возможностью формирования первого начального значения для использования при дешифровании по меньшей мере следующего кодированного и шифрованного блока данных, или пакета данных, или потока данных из упомянутого множества кодированных и шифрованных блоков данных, или пакетов данных, или потоков данных, чтобы получить по меньшей мере следующий кодированный блок данных, или пакет данных, или поток данных, и

декодирования упомянутого следующего кодированного блока данных, или пакета данных, или потока данных, чтобы получить следующий декодированный блок данных, или пакет данных, или поток данных для включения в декодированные данные (D3); и

(iii) упомянутая схема обработки данных выполнена с возможностью формирования следующего начального значения для использования при дешифровании и декодировании последующего кодированного и шифрованного блока данных, или пакета данных, или потока данных из упомянутого множества кодированных и шифрованных блоков данных, или пакетов данных, или потоков данных, и так последовательно и повторно до тех пор, пока упомянутое множество кодированных и шифрованных блоков данных, или пакетов данных, или потоков данных не будет дешифровано и декодировано в декодированные данные (D3),

при этом для данного кодированного и шифрованного блока данных, или пакета данных, или потока данных начальное значение формируют на основании предшествующего ему декодированного блока данных, или пакета данных, или потока данных.

24. Декодер (120) по п. 23, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью многократного использования упомянутого по меньшей мере одного ключа в комбинации с упомянутыми начальными значениями для дешифрования упомянутого множества кодированных и шифрованных блоков данных, или пакетов данных, или потоков данных.

25. Декодер (120) по п. 23 или 24, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью использования упомянутого по меньшей мере одного ключа для дешифрования только упомянутого первого кодированного и шифрованного блока данных, или пакета данных, или потока данных.

26. Декодер (120) по любому из пп. 23-25, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью применения инициализационного вектора (IV) в комбинации с упомянутым по меньшей мере одним ключом при дешифровании упомянутого первого кодированного блока данных, или пакета данных, или потока данных.

27. Декодер (120) по любому из пп. 23-26, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью дешифрования кодированных и шифрованных данных (E2) для формирования соответствующих декодированных данных (D3), последовательно и повторно, с ветвлением на множество параллельных последовательностей операций по декодированию и дешифрованию шифрованных и кодированных блоков данных, или пакетов данных, или потоков данных с использованием соответствующих начальных значений.

28. Декодер (120) по любому из пп. 23-27, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью дешифрования и декодирования кодированных и шифрованных данных (E2), представленных по меньшей мере в одной из следующих форм: кодированные и шифрованные одномерные данные, кодированные и шифрованные многомерные данные, кодированные и шифрованные текстовые данные, кодированные и шифрованные двоичные данные, кодированные и шифрованные данные датчиков, кодированные и шифрованные аудиоданные, кодированные и шифрованные данные изображений, кодированные и шифрованные видеоданные.

29. Декодер (120) по любому из пп. 23-28, отличающийся тем, что упомянутая схема обработки данных выполнена с возможностью обеспечения приема упомянутого по меньшей мере одного ключа в декодере (120) для использования при последующем

дешифровании и декодировании кодированных и шифрованных данных (E2), вручную или при помощи шифрованного сообщения электронной почты, или по шифрованному соединению связи.

30. Декодер (120) по п. 29, отличающийся тем, что упомянутое шифрованное соединение связи реализовано при помощи протокола защищенных сокетов (SSL) / безопасности транспортного уровня (TLS).

31. Декодер (120) по любому из пп. 23-30, отличающийся тем, что упомянутая схема обработки данных реализована с применением по меньшей мере одного процессора с архитектурой с ограниченным набором команд (RISC), который выполнен с возможностью исполнения программных инструкций.

32. Способ дешифрования и декодирования кодированных и шифрованных данных (120), включающих множество кодированных и шифрованных блоков данных, или пакетов данных, или потоков данных, при помощи декодера (120), при этом декодер (120) включает схему обработки данных для обработки кодированных и шифрованных данных (E2) для формирования соответствующих декодированных данных (D3), и в декодер (120), при его функционировании, предоставляют по меньшей мере один ключ, который используют при формировании декодированных данных (D3), при этом упомянутая схема обработки данных объединяет процессы декодирования и дешифрования для формирования декодированных данных (D3), и способ включает:

(i) дешифрование по меньшей мере первого кодированного и шифрованного блока данных, или пакета данных, или потока данных из упомянутого множества кодированных и шифрованных блоков данных, или пакетов данных, или потоков данных с использованием упомянутого по меньшей мере одного ключа, чтобы сформировать по меньшей мере первый кодированный блок данных, или пакет данных, или поток данных;

(ii) декодирование упомянутого по меньшей мере первого блока данных, или пакета данных, или потока данных, чтобы получить первый декодированный блок данных, или пакет данных, или поток данных для включения в декодированные данные (D3);

(iii) формирование первого начального значения для использования при дешифровании и декодировании следующего кодированного и шифрованного блока данных, или пакета данных, или потока данных из упомянутого множества кодированных и шифрованных блоков данных, или пакетов данных, или потоков данных, чтобы получить по меньшей мере следующий декодированный блок данных, или пакет данных, или поток данных для включения в декодированные данные (D3); и

(iv) формирование следующего начального значения для использования при дешифровании и декодировании последующего кодированного и шифрованного блока данных, или, пакета данных, или потока данных из упомянутого множества кодированных и шифрованных блоков данных, или пакетов данных, или потоков данных, и так последовательно и повторно до тех пор, пока упомянутое множество кодированных и шифрованных блоков данных, или пакетов данных, или потоков данных не будет дешифровано и декодировано в декодированные данные (D3),

при этом для данного кодированного и шифрованного блока данных, или пакета данных, или потока данных начальное значение формируют на основании предшествующего ему декодированного блока данных, или пакета данных, или потока данных.

33. Способ по п. 32, включающий многократное использование упомянутой схемой обработки данных упомянутого по меньшей мере одного ключа в комбинации с

упомянутыми начальными значениями для дешифрования упомянутого множества кодированных и шифрованных блоков данных, или пакетов данных, или потоков данных.

34. Способ по п. 32 или 33, включающий использование упомянутой схемой обработки данных упомянутого по меньшей мере одного ключа для дешифрования только упомянутого первого кодированного и шифрованного блока данных, или пакета данных, или потока данных.

35. Способ по любому из пп. 32-34, включающий выполнение упомянутой схемы обработки данных с возможностью применения инициализационного вектора (IV) в комбинации с упомянутым по меньшей мере одним ключом при дешифровании упомянутого по меньшей мере первого кодированного и шифрованного блока данных, или пакета данных, или потока данных.

36. Способ по любому из пп. 32-35, включающий выполнение, упомянутой схемой обработки данных, дешифрования кодированных и шифрованных данных (E2) для формирования соответствующих декодированных данных (D3), последовательно и повторно, с ветвлением на множество параллельных последовательностей операций по декодированию и дешифрованию шифрованных и кодированных блоков данных, или пакетов данных, или потоков данных с использованием соответствующих начальных значений.

37. Способ по любому из пп. 32-36, включающий выполнение, упомянутой схемой обработки данных, дешифрования и декодирования кодированных и шифрованных данных (E2), представленных по меньшей мере в одной из следующих форм: кодированные и шифрованные одномерные данные, кодированные и шифрованные многомерные данные, кодированные и шифрованные текстовые данные, кодированные и шифрованные двоичные данные, кодированные и шифрованные данные датчиков, кодированные и шифрованные аудиоданные, кодированные и шифрованные данные изображений, кодированные и шифрованные видеоданные.

38. Способ по любому из пп. 32-37, включающий выполнение, упомянутой схемой обработки данных, приема упомянутого по меньшей мере одного ключа в декодере (120) для использования при последующем дешифровании и декодировании кодированных и шифрованных данных (E2), вручную или при помощи шифрованного сообщения электронной почты, или по шифрованному соединению связи.

39. Способ по п. 38, отличающийся тем, что упомянутое шифрованное соединение связи реализуют при помощи протокола защищенных сокетов (SSL) / безопасности транспортного уровня (TLS).

40. Способ по любому из пп. 32-39, отличающийся тем, что упомянутую схему обработки данных реализуют с применением по меньшей мере одного процессора с архитектурой с ограниченным набором команд (RISC), который выполнен с возможностью исполнения программных инструкций.

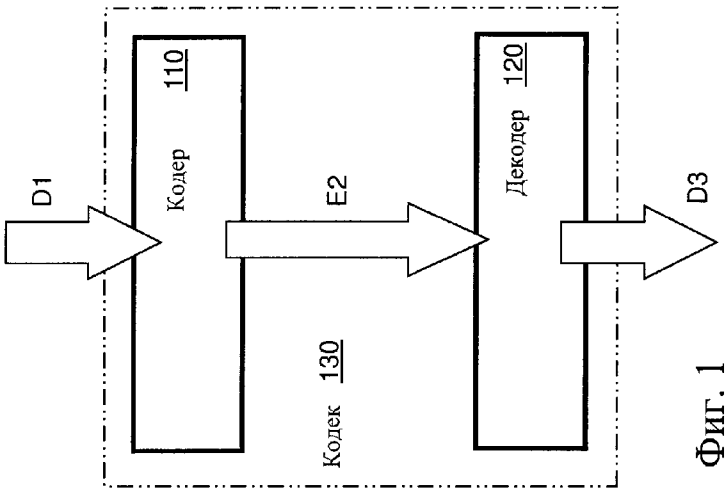
41. Кодек (130), включающий по меньшей мере один кодер (110) по п. 1 и по меньшей мере один декодер (120) по п. 23.

42. Компьютерный программный продукт, включающий машиночитаемый носитель, имеющий хранимые на нем машиночитаемые инструкции, исполняемые при помощи компьютеризованного устройства, включающего процессорную аппаратуру, для исполнения способа по п. 12 или 32.

1

Кодер, декодер и способ

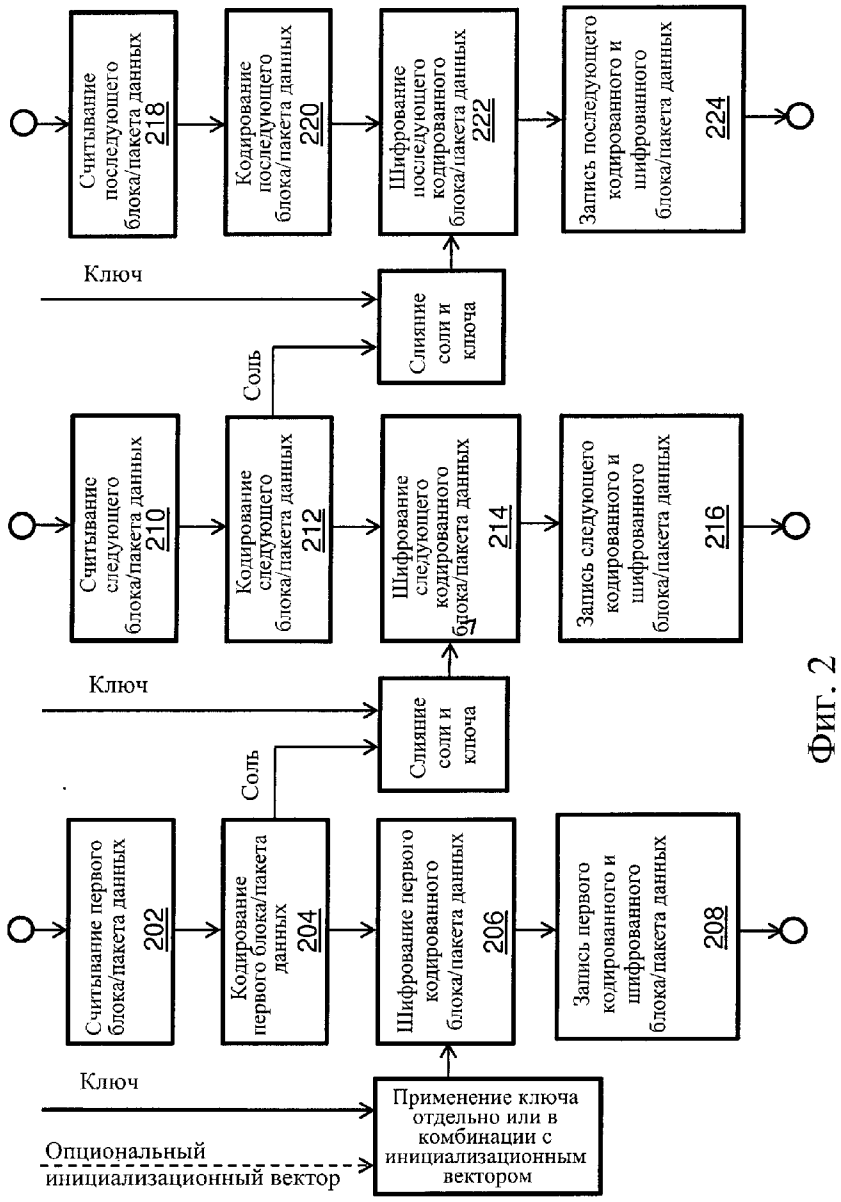
1/5



Фиг. 1

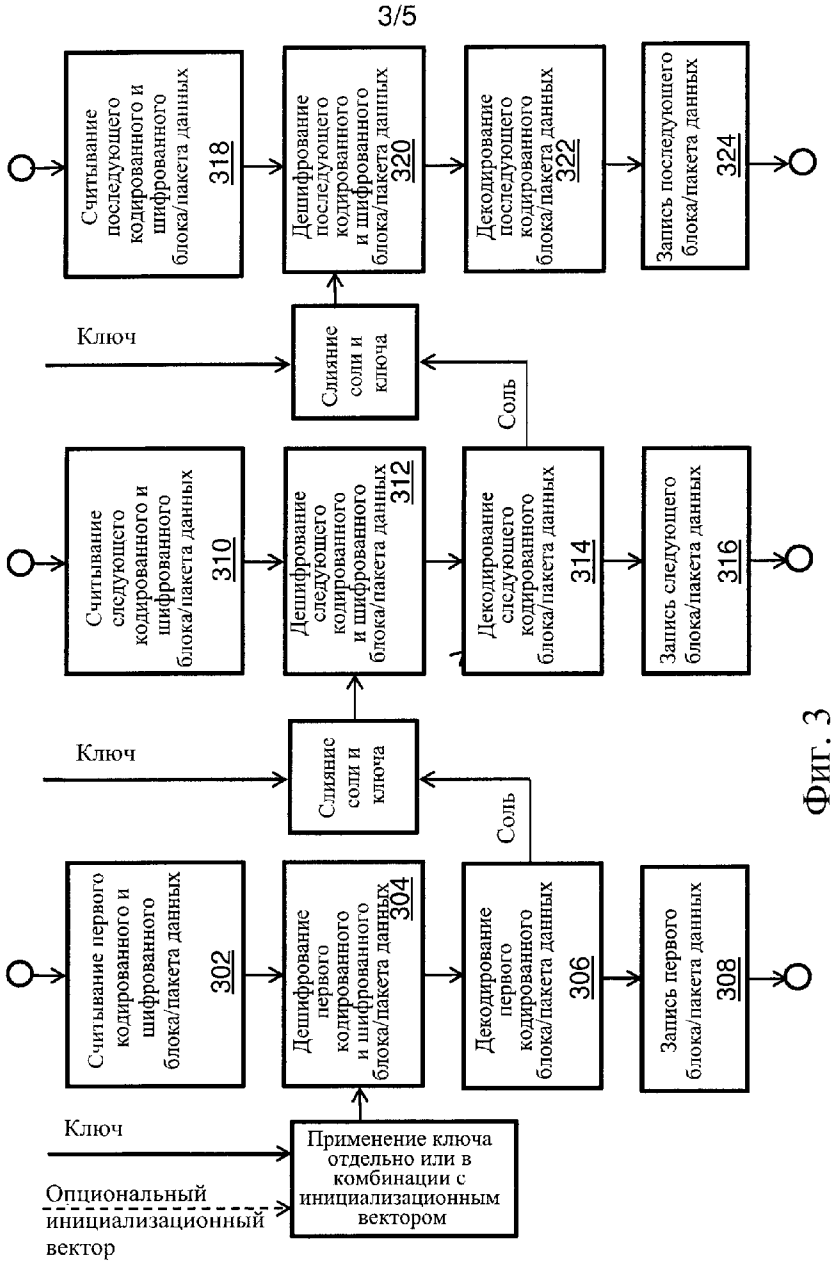
2

Кодер, декодер и способ



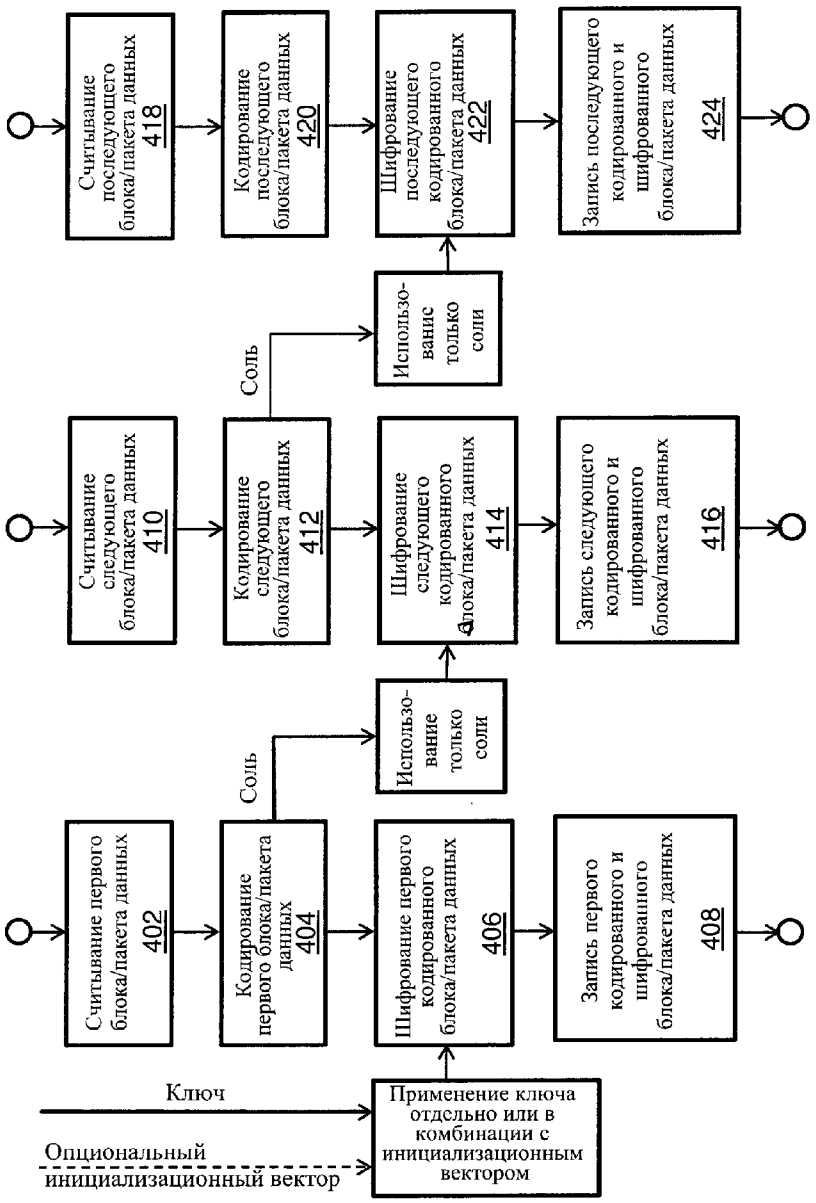
Фиг. 2

3

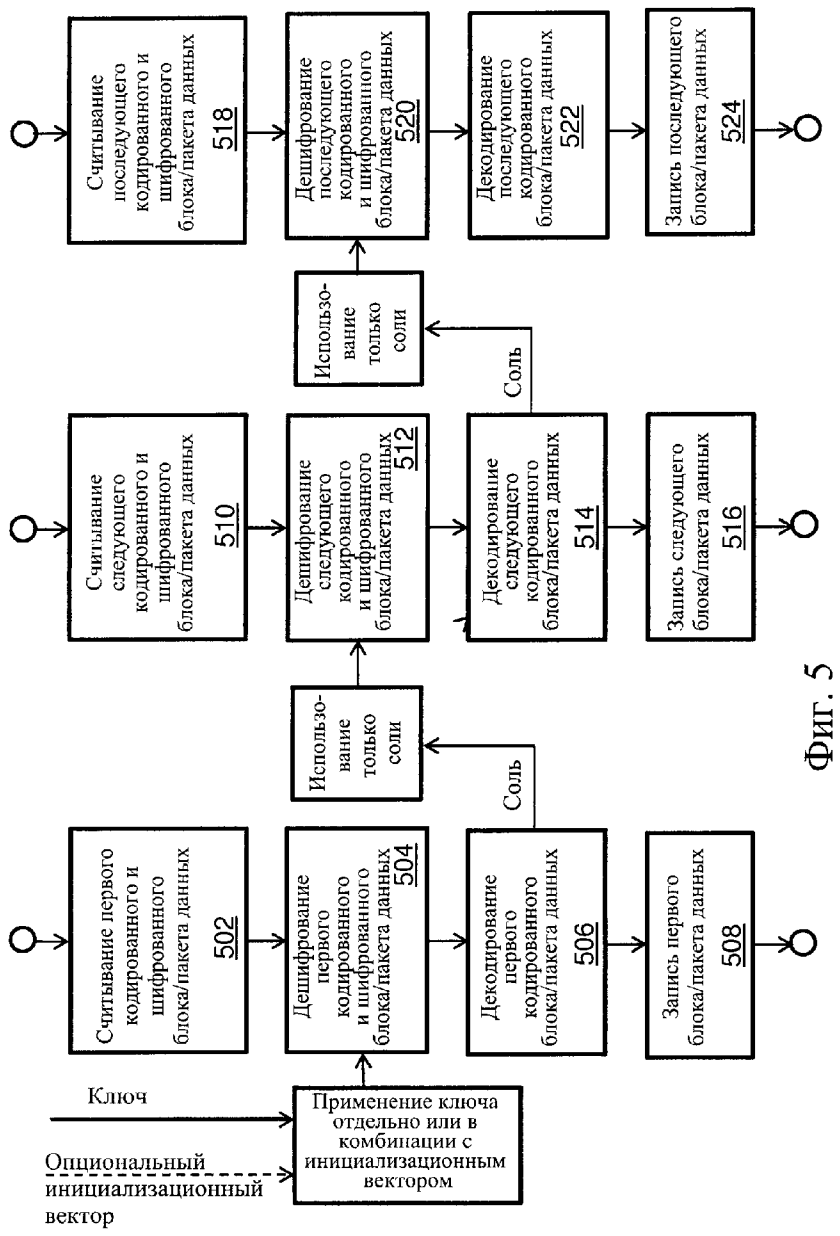

Считывание первого кодированного и шифрованного блока/пакета данных 302
Дешифрование первого кодированного и шифрованного блока/пакета данных 304
Декодирование первого кодированного блока/пакета данных 306
Запись первого блока/пакета данных 308
Ключ
Опциональный инициализационный вектор
Применение ключа отдельно или в комбинации с инициализационным вектором
Считывание следующего кодированного и шифрованного блока/пакета данных 310
Дешифрование следующего кодированного и шифрованного блока/пакета данных 312
Декодирование следующего кодированного блока/пакета данных 314
Запись следующего блока/пакета данных 316
Ключ
Слияние соли и ключа
Соль
Считывание последующего кодированного и шифрованного блока/пакета данных 318
Дешифрование последующего кодированного и шифрованного блока/пакета данных 320
Декодирование последующего кодированного блока/пакета данных 322
Запись последующего блока/пакета данных 324
Ключ
Слияние соли и ключа
Соль

Фиг. 3

4


Считывание первого блока/пакета данных 402
Кодирование первого блока/пакета данных 404
Шифрование первого кодированного блока/пакета данных 406
Запись первого кодированного и шифрованного блока/пакета данных 408
Соль
Использование только соли
Считывание следующего блока/пакета данных 410
Кодирование следующего блока/пакета данных 412
Шифрование следующего кодированного блока/пакета данных 414
Запись следующего кодированного и шифрованного блока/пакета данных 416
Соль
Использование только соли
Считывание последующего блока/пакета данных 418
Кодирование последующего блока/пакета данных 420
Шифрование последующего кодированного блока/пакета данных 422
Запись последующего кодированного и шифрованного блока/пакета данных 424
Ключ
Опциональный инициализационный вектор
Применение ключа отдельно или в комбинации с инициализационным вектором

Фиг. 4


Считывание первого кодированного и шифрованного блока/пакета данных 502
Дешифрование первого кодированного и шифрованного блока/пакета данных 504
Декодирование первого кодированного блока/пакета данных 506
Запись первого блока/пакета данных 508
Использование только соли
Соль
Считывание следующего кодированного и шифрованного блока/пакета данных 510
Дешифрование следующего кодированного и шифрованного блока/пакета данных 512
Декодирование следующего кодированного блока/пакета данных 514
Запись следующего блока/пакета данных 516
Использование только соли
Соль
Считывание последующего кодированного и шифрованного блока/пакета данных 518
Дешифрование последующего кодированного и шифрованного блока/пакета данных 520
Декодирование последующего кодированного блока/пакета данных 522
Запись последующего блока/пакета данных 524
Ключ
Опциональный инициализационный вектор
Применение ключа отдельно или в комбинации с инициализационным вектором

Фиг. 5