РОССИЙСКАЯ ФЕДЕРАЦИЯ

(19) **RU** (11) **2 636 109**(13) **C2**

(51) МПК
H04L 9/08 (2006.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: **2015129778, 20.12.2013**

(24) Дата начала отсчета срока действия патента:
20.12.2013

Дата регистрации:
20.11.2017

Приоритет(ы):
(30) Конвенционный приоритет:
21.12.2012 EP 12198794.5;
21.12.2012 US 61/740,488

(43) Дата публикации заявки: **27.01.2017** Бюл. № 3

(45) Опубликовано: **20.11.2017** Бюл. № 32

(85) Дата начала рассмотрения заявки PCT на национальной фазе: **21.07.2015**

(86) Заявка PCT:
EP 2013/077842 (20.12.2013)

(87) Публикация заявки PCT:
WO 2014/096420 (26.06.2014)

Адрес для переписки:
129090, Москва, ул. Б. Спасская, 25, стр. 3, ООО "Юридическая фирма Городисский и Партнеры"

(72) Автор(ы):
ГОМЕС Доминго (NL),
ГАРСИЯ МОРЧОН Оскар (NL),
ТОЛХЭЙЗЕН Людовикус Маринус Герардус Мария (NL),
ГУТЬЕРРЕС Хайме (NL)

(73) Патентообладатель(и):
КОНИНКЛЕЙКЕ ФИЛИПС Н.В. (NL)

(56) Список документов, цитированных в отчете о поиске: **WO 2007/149850 A2, 27.12.2007. US 2009/0129599 A1, 21.05.2009. WO 2010/032161 A1, 25.03.2010. WO 2010/106496 A1, 23.09.2010. RU 2385539 C1, 27.03.2010.**

(54) ИСПОЛЬЗУЮЩЕЕ ОБЩИЙ КЛЮЧ СЕТЕВОЕ УСТРОЙСТВО И ЕГО КОНФИГУРИРОВАНИЕ

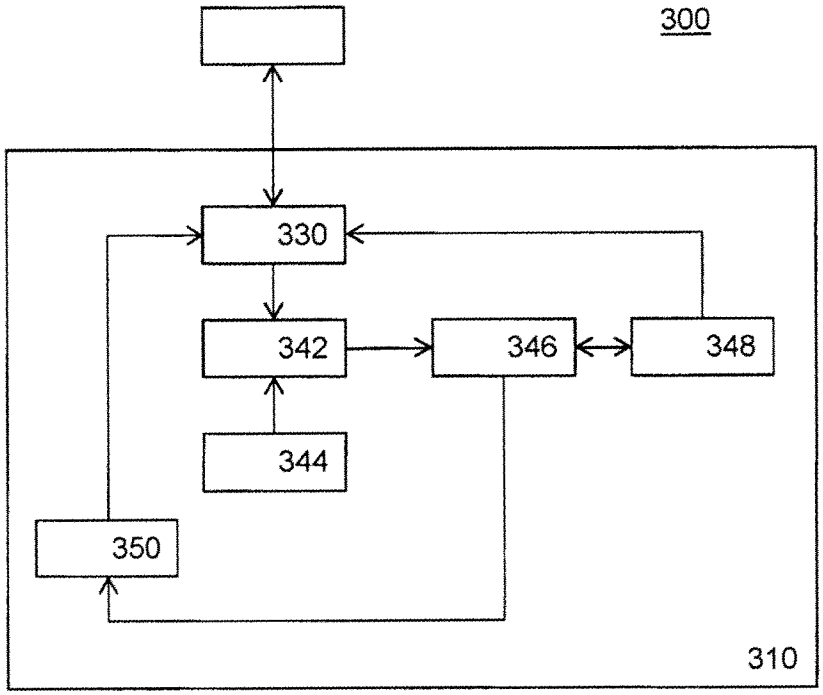
(57) Реферат:

Изобретение относится к области конфигурирования сетевых устройств. Технический результат – обеспечение эффективной сетевой безопасности. Способ конфигурирования сетевого устройства для использования общего ключа содержит этапы, на которых получают в электронной форме по меньшей мере два набора параметров, набор параметров, содержащий частный модуль (p_1), публичный модуль (N) и двумерный полином (f_1), имеющий целые коэффициенты, бинарное представление публичного модуля и бинарное представление частного модуля являются одинаковыми в имеющих по меньшей мере длину ключа (b) последовательных разрядах, генерируют материал локального ключа для сетевого устройства, этап генерирования содержит этапы, на которых получают в электронной форме идентификационный номер (A) для сетевого устройства и для каждого набора параметров из по меньшей мере двух наборов параметров получают соответствующий одномерный полином посредством определения, используя устройство манипулирования полиномом, одномерного полинома из двумерного полинома набора параметров посредством подстановки идентификационного номера в упомянутый двумерный полином и приведения результата подстановки по модулю частного модуля набора параметров и

RU 2636109 C2

RU 2636109 C2

электронным образом сохраняют на сетевом устройстве сгенерированный материал локального ключа, содержащий публичный модуль каждого набора параметров и соответствующий одномерный полином каждого набора параметров. 5 н. и 12 з.п. ф-лы, 6 ил.



ФИГ.3

RU 2636109 C2

RU 2636109 C2

RUSSIAN FEDERATION

(19) **RU** (11) **2 636 109**(13) **C2**

(51) Int. Cl.
H04L 9/08 (2006.01)

FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(12) ABSTRACT OF INVENTION

(21)(22) Application: **2015129778, 20.12.2013**

(24) Effective date for property rights:
20.12.2013

Registration date:
20.11.2017

Priority:
(30) Convention priority:
21.12.2012 EP 12198794.5;
21.12.2012 US 61/740,488

(43) Application published: **27.01.2017** Bull. № 3

(45) Date of publication: **20.11.2017** Bull. № 32

(85) Commencement of national phase: **21.07.2015**

(86) PCT application:
EP 2013/077842 (20.12.2013)

(87) PCT publication:
WO 2014/096420 (26.06.2014)

Mail address:
129090, Moskva, ul. B. Spasskaya, 25, str. 3, OOO "Yuridicheskaya firma Gorodisskij i Partnery"

(72) Inventor(s):
GOMES Domingo (NL),
GARSIYA MORCHON Oskar (NL),
TOLKHEJZEN Lyudovikus Marinus Gerardus Mariya (NL),
GUTERRES Khajme (NL)

(73) Proprietor(s):
KONINKLEJKE FILIPS N.V. (NL)

RU 2636109 C2

(54) USING GENERAL KEY NETWORKING DEVICE AND ITS CONFIGURATION

(57) Abstract:

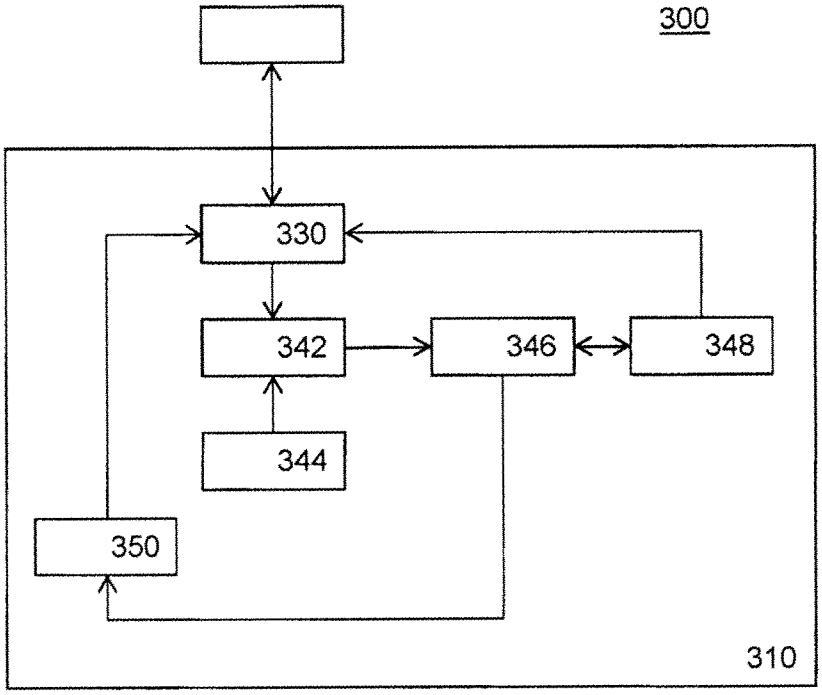
FIELD: information technology.

SUBSTANCE: method for configuring a network device for using a public key comprises the steps of receiving at least two sets of parameters in electronic form, a set of parameters comprising a private module (p_1), a public module (N), and a two-dimensional polynomial (f_1) having integer coefficients, a binary representation of public module and a binary representation of private module are the same in consecutive bits having at least the length of the key (b), generating a local key material for the network device, the generating step comprising steps of obtaining identification number (A) for the network device and for each set of parameters from at least two sets of parameters obtaining the corresponding one-dimensional polynomial by determining, using the device of manipulating the polynomial, the one-dimensional polynomial from two-dimensional polynomial parameter set by substituting the ID number in the said two-dimensional polynomial and cast the result of substitution in modulus of private module parameter set, and generated local key material containing the module of each set of parameters and corresponding one-dimensional polynomial for each set of parameters are electronically stored on a network device.

EFFECT: ensuring effective network security.

17 cl, 6 dwg

RU 2636109 C2



ФИГ.3

RU 2636109 C2

RU 2636109 C2

ОБЛАСТЬ ТЕХНИКИ, К КОТОРОЙ ОТНОСИТСЯ ИЗОБРЕТЕНИЕ

Изобретение относится к способу конфигурирования сетевого устройства для использования общего ключа, способ содержит генерирование материала локального ключа для сетевого устройства, содержит получение в электронной форме идентификационного номера сетевого устройства, определение, используя устройство манипулирования полиномом, одномерного полинома из двумерного полинома посредством подстановки идентификационного номера в упомянутый двумерный полином, и электронное сохранение сгенерированного материала локального ключа на сетевом устройстве.

Изобретение дополнительно относится к способу определения общего ключа для первого сетевого устройства, причем ключ является криптографическим ключом, способ содержит получение материала локального ключа для первого сетевого устройства в электронной форме, материал локального ключа содержит одномерный полином, получение идентификационного номера для второго сетевого устройства, причем второе сетевое устройство отличается от первого сетевого устройства, подстановку идентификационного номера второго сетевого устройства в одномерный полином, и получение из него общего ключа.

Изобретение дополнительно относится к системе для конфигурирования сетевого устройства для использования общего ключа и к сетевому устройству, сконфигурированному, чтобы определять общий ключ.

УРОВЕНЬ ТЕХНИКИ ИЗОБРЕТЕНИЯ

Статья, написанная SONG GUO и др.: «Основанная на перестановках много-полиномиальная схема для попарного установления ключа в сенсорных сетях (A Permutation-Based Multi-Polynomial Scheme for Pairwisc Key Establishment in Sensor Networks) », COMMUNICATIONS (ICC), международная конференция IEEE (Институт инженеров по электротехнике и радиоэлектронике) 2010 года, IEEE, Пискатауэй, Нью-Джерси, США, 23 мая 2010 года (2010-05-23), страницы 1-5, раскрывает решение предшествующего уровня техники.

Если имеется сеть связи, содержащая множество сетевых устройств, существует проблема установления безопасных соединений между парами таких сетевых устройств. Один из способов осуществления этого описан в работе C. Blundo, A. Dc Santis, A. Herzberg, S. Kuttcn, U. Vaccaro и M. Yung, «Идеально безопасное распределение ключей для динамических конференций (Perfectly-Secure Key distribution for Dynamic Conferences) », Springer, записи лекций по математике, том 740, страницы 471-486, 1993 год (указываемая ссылкой, как 'Blundo').

Предполагается наличие центральной службы, также указываемой как сетевая служба или как доверенная третья сторона (Trusted Third Party,TTP), которая генерирует симметричный двумерный полином f(x,y) с коэффициентами в конечном поле F с p элементами, где p - простое число или степень простого числа. Каждое устройство имеет идентификационный номер в F и обеспечивается материалом локального ключа посредством TTP. Для устройства с идентификатором η, материал локального ключа является коэффициентами полинома f(η,y).

Если устройство η хочет связаться с устройством η', оно использует свой материал ключа, чтобы сгенерировать ключ K(η,η')=f(η,η'). Так как f симметричен, генерируется одинаковый ключ.

Проблема этой схемы использования общего ключа возникает, если взломщик знает материал ключа t+1 или более устройств, где t - степень двумерного полинома. Взломщик может восстановить полином f(x,y). В этот момент безопасность системы полностью

взломана. При наличии идентификационных номеров любых двух устройств взломщик может восстановить общий ключ этой пары устройств.

СУЩНОСТЬ ИЗОБРЕТЕНИЯ

Было бы полезно иметь улучшенный способ для установления общего ключа между двумя сетевыми устройствами. Изобретение определено независимыми пунктами формулы изобретения; зависимые пункты определяют преимущественные варианты осуществления. Предоставляется способ конфигурирования сетевого устройства для использования общего ключа и способ определения общего ключа для сетевого устройства.

Способ конфигурирования сетевого устройства для использования общего ключа содержит получение в электронной форме по меньшей мере двух наборов параметров, причем набор параметров содержит частный модуль, публичный модуль и двумерный полином, имеющий целые коэффициенты, бинарное представление публичного модуля и бинарное представление частного модуля являются одинаковыми в последовательных разрядах по меньшей мере длины ключа, генерирование материала локального ключа для сетевого устройства, содержащее получение в электронной форме идентификационного номера для сетевого устройства, и получение для каждого набора параметров из по меньшей мере двух наборов параметров соответствующего одномерного полинома посредством: определения, используя устройство манипулирования полиномом, одномерного полинома из двумерного полинома набора параметров посредством подстановки идентификационного номера в упомянутый двумерный полином, и приведения результата подстановки по модулю частного модуля набора параметров, и электронное сохранение на сетевом устройстве сгенерированного материала локального ключа, содержащего публичный модуль каждого набора параметров и соответствующий одномерный полином каждого набора параметров.

Способ определения общего ключа для первого сетевого устройства, ключ является криптографическим ключом, содержащий получение материала локального ключа для первого сетевого устройства в электронной форме, материал локального ключа содержит по меньшей мере два, необязательно запутанных одномерных полинома и соответствующий публичный модуль, получение идентификационного номера для второго сетевого устройства, причем второе сетевое устройство отличается от первого сетевого устройства, для каждого из по меньшей мере двух, необязательно запутанных одномерных полиномов, подстановку идентификационного номера второго сетевого устройства в упомянутый одномерный полином, и приведение результата подстановки по модулю публичного модуля, соответствующего упомянутому одномерному полиному, и сложение вместе результатов приведений по модулю публичного модуля и приведение по модулю ключевого модуля, и получение общего ключа из результата приведения по модулю ключевого модуля.

В варианте осуществления способ содержит приведение результата подстановки по модулю публичного модуля, деление результата на степень двойки и приведение по модулю ключевого модуля.

Любая пара из двух сетевых устройств из множества сетевых устройств, каждое из которых имеет идентификационный номер и материал локального ключа, сгенерированный для идентификационного номера, могут выработать общий ключ с помощью нескольких ресурсов. Двум сетевым устройствам надо лишь обменяться своими идентификационными номерами, которые не нужно держать в секрете, и выполнить полиномиальные вычисления. Тип необходимых вычислений не требует больших вычислительных ресурсов, что означает, что этот способ пригоден для

малозатратных типов применений с большим объемом.

Материал локального ключа был получен из общего полинома в материале корневого ключа; это позволяет обоим сетевым устройствам в паре сетевых устройств получить одинаковый общий ключ. Если все двумерные полиномы симметричны, тогда любые два сетевых устройства могут получить общий полином. Если некоторые или все двумерные полиномы ассиметричны, некоторые пары устройств могут, а некоторые не могут получить общий ключ.

Материал локального ключа получается из набора параметров, в частности, из множества разных публичных модулей и множества двумерных полиномов. Полученный в результате материал локального ключа содержит множество, обычно разных, одномерных полиномов, каждый с соответствующим публичным модулем.

Если был использован только один набор параметров, тогда сетевое устройство обеспечивается коэффициентами полинома так, что посредством оценивания его по модулю N и взятия b разрядов оно может сгенерировать b-разрядный ключ с любым другим устройством. Это относится к задаче шумной полиномиальной интерполяции, то есть при наличии большого количества таких b-разрядных ключей, взломщик может восстановить полином заданного атакуемого объекта.

Например, атака, сталкивающаяся с системой с одним набором параметров, могла бы получить эти b-разрядные значения посредством следующих 2 этапов: взломщик компрометирует N_c устройств, связанных с N_c материалами ключа, и взломщик использует эти N_c материалов ключа, чтобы получить N_c b-разрядных ключей (посредством оценивания каждого из материалов ключа в идентификаторе атакуемого устройства). Это означает, что развитие, сделанное в задаче шумной полиномиальной интерполяции, может быть расширено на атаки системы с одним набором параметров. Это считается нежелательным.

Наличие множества наборов параметров избегает этой проблемы посредством смешивания модульных операций на устройстве, а также во время генерирования локального ключа.

Общий ключ K_{AB} между парой устройств A и B получается, как сумма по меньшей мере двух (в общем случае m') подключей K_{AB}^{i}, то есть $K_{AB} = K_{AB}^{1} + K_{AB}^{2}$. Каждый подключ K_{AB}^{i} сгенерирован из отдельного материала ключа, в котором модульные операции выполняются по модулю публичного модуля N_i. Так как модульные операции смешиваются во время генерирования локального ключа, а также во время генерирования общего ключа, невозможно расширить атаки с использованием шумной полиномиальной интерполяции на эту криптографическую систему. Даже если взломщик получит доступ к N_c b-разрядным ключам, каждый из них получен из двух подключей, каждый подключ, вытекающий из оценивания отдельного материала ключа. Но взломщик не сможет различить подключи, поэтому взломщик не может восстановить два (в общем случае m') материала ключа атакуемого устройства.

Имеется два уровня важности атаки на устройства. В более низкой важности взломщик получает доступ только ко многим общим ключам. В более высокой важности взломщик получает доступ ко многим материалам локального ключа. Оказывается, наличие смешивающих модульных операций на сетевом устройстве является хорошей

мерой против атаки более низкой важности. Тем не менее, если взломщик имеет доступ к самому материалу ключа, тогда он также имеет доступ к подключам.

Последней проблемы избегают посредством добавления шума к двум материалам ключа устройства. Добавление запутывающего числа к материалу локального ключа нарушает связь между материалом локального ключа и материалам корневого ключа. Связь, которая присутствовала бы между незапутанным одномерным полиномом и (симметричными) двумерными полиномами, теперь отсутствует. Это означает, что прямая атака такой схемы больше не работает.

Что интересно, добавление шума к двум материалам ключа устройства, такого, что добавленный шум равняется нулю по модулю 2^b, дополнительно улучшает систему. В этом случае: сгенерированные ключи все еще шумные, и, таким образом, взломщик не может использовать их для восстановления долей материала ключа атакуемого устройства; к тому же, чтобы удалить шум, взломщику необходимо сложить их, но тогда он получит суммарное значение, как выше, и не сможет различить компоненты, возникающие из каждого из материалов ключа. Эта методика может быть легко обобщена на любое количество материалов ключа. Это условие также может быть расширено, чтобы гарантировать, что шум равняется нулю в b разрядах, расположенных не в самых младших разрядах, а в каком-то другом месте.

В варианте осуществления бинарное представление всего публичного модуля и бинарное представление частного модуля в каждом наборе параметров являются одинаковыми в имеющих по меньшей мере длину ключа (b) последовательных разрядах. Отметим, что может использоваться множество частных модулей; они могут выбираться так, чтобы бинарное представление любого из множества частных модулей публичного модуля и бинарное представление частного модуля были одинаковыми в имеющих по меньшей мере длину ключа (b) последовательных разрядах. Для каждого частного модуля из множества частных модулей выбирается, необязательно, симметричный, двумерный полином, имеющий целые коэффициенты, чтобы получить множество, необязательно, симметричных, двумерных полиномов.

Так как получение материала локального ключа использует частный модуль, который отличается от публичного модуля, математическая связь, которая присутствовала бы при работе, скажем, в одном конечном поле, нарушается. Это означает, что обычные математические инструменты для анализа полиномов, например конечномерная алгебра, больше не применимы. В лучшем случае взломщик может использовать намного менее эффективные конструкции, такие как решетки. Кроме того, при получении общего ключа объединяются две операции по модулю, которые не являются совместимыми в обычном математическом смысле; поэтому математическая конструкция избегается в двух местах. Способ обеспечивает прямое попарное генерирование ключа и устойчив к захвату очень большого числа, например порядка 10^5 или даже выше, сетевых устройств. С другой стороны, так как частный и публичный модуль перекрываются в некотором количестве последовательных разрядов, два сетевых устройства, которые имеют материал локального ключа, с большой вероятностью могут получить одинаковый общий ключ.

Особой догадкой изобретателей было то, что публичный модуль не должен быть простым числом. В одном из вариантов осуществления публичный модуль является составным. Кроме того, нет причины, по которой публичный модуль обязан быть числом со 'всеми единичными' разрядами, например числом, которое состоит только из разрядов ‘1’ в своем бинарном представлении. В варианте осуществления публичный модуль не является степенью двойки минус один. В варианте осуществления бинарное

представление публичного модуля содержит по меньшей мере один нулевой разряд (не считая ведущие нули, то есть бинарное представление публичного модуля содержит по меньшей мере один нулевой разряд, который младше самого старшего разряда публичного модуля). В варианте осуществления публичный модуль является степенью двойки минус один и составным числом.

В варианте осуществления публичный модуль одного или более наборов параметров больше, чем один или более частных модулей.

В варианте осуществления все имеющие по меньшей мере длину ключа последовательные разряды бинарного представления разницы публичного модуля и частного модуля являются нулевыми разрядами. Эта разница должна оцениваться, используя представление числа со знаком разницы публичного модуля и частного модуля, а не представление с двумя дополнениями. В качестве альтернативы можно потребовать, чтобы все имеющие по меньшей мере длину ключа последовательные разряды бинарного представления абсолютного значения разницы публичного модуля и частного модуля являлись нулевыми разрядами. Имеется набор имеющих длину ключа (b) последовательных позиций, в которых бинарное представление публичного модуля согласуется с бинарным представлением всех частных модулей.

Последовательные позиции разрядов, в которых бинарное представление публичного модуля согласуется с частными модулями, могут являться самыми младшими разрядами. В варианте осуществления все самые младшие имеющие длину ключа разряды бинарного представления разницы публичного модуля и частного модуля являются нулевыми разрядами; это имеет преимущество, которое состоит в том, что деление на степень двойки не является необходимым при получении общего ключа.

В варианте осуществления во всех наборах параметров, одинаковые по меньшей мере имеющие длину ключа (b) последовательные разряды бинарного представления публичного модуля соответствующего набора параметров такие же, как и самые младшие имеющие длину ключа (b) разряды частного модуля соответствующего набора параметров. То есть имеется набор последовательных позиций разрядов, который, в каждом наборе параметров, указывает, где согласуются публичный и частный модули. Хотя этот набор последовательных позиций разрядов одинаков для всех наборов параметров, сами разряды могут отличаться для разных наборов параметров. В варианте осуществления имеющие по меньшей мере длину ключа (b) последовательные разряды являются самыми младшими имеющими длину ключа (b) разрядами. То есть набор позиций разрядов является позициями самых младших разрядов.

Допускается, что один частный модуль из множества частных модулей равняется публичному модулю. Однако если используется только один частный модуль, это является нежелательным.

Желательно, чтобы частные модули вводили существенную нелинейность. В варианте осуществления имеется набор последовательных позиций разрядов, в которых публичный модуль отличается от каждого частного модуля. Более того, также можно наложить условие, чтобы частные модули отличались друг от друга; попарное сравнение бинарного представления частного модуля также может отличаться в по меньшей мере одном разряде в наборе, скажем, имеющих по меньшей мере длину ключа последовательных разрядов, набор, равный для всех частных модулей и, возможно, также для публичного модуля.

Сетевое устройство может являться электронным устройством, оборудованным электронным средством связи и вычислительным средством. Электронное устройство может прикрепляться, например, в форме радиометки, к любому неэлектронному

объекту. Например, этот способ был бы пригоден для ‘интернета вещей’. Например, объекты, в частности недорогие объекты, могут быть оборудованы радиометками, через которые они могут взаимодействовать, например могут быть идентифицированы.

Такие объекты могут инвентаризироваться с помощью электронного средства, такого как компьютер. Украденные или поломанные элементы могут быть легко отслежены и найдены. Одним особо многообещающим применением является лампа, содержащая сетевое устройство, сконфигурированное, чтобы определять общий ключ. Такая лампа может безопасно передавать свой статус; такой лампой можно безопасно манипулировать, например вкручивать и/или выкручивать. Сетевое устройство может являться одним из множества сетевых устройств, каждое из которых содержит электронное устройство связи для передачи и приема идентификационного номера, и для передачи и приема сообщения о статусе, и каждое из них содержит интегральную схему, сконфигурированную для получения общего ключа, следуя способу согласно изобретению.

В варианте осуществления способ согласно изобретению может использоваться в качестве криптографического способа для протоколов безопасности, таких как IPSec, (D)TLS, HIP, или ZigBee. В частности, устройство, использующее один из этих протоколов, ассоциируется с идентификатором. Второе устройство, желающее взаимодействовать с первым устройством, может сгенерировать общий парный ключ с первым устройством при наличии его идентификатора, и парный ключ (или ключ, полученный из него посредством, например, функции получения ключа) может использоваться в способе согласно вышеописанным протоколам на основании предварительного общего ключа. В частности, идентификатор устройства, как он определен в данном изобретении, может являться сетевым адресом, таким как короткий адрес ZigBee, IP-адрес или идентификатор хоста. Идентификатор также может являться IEEE адресом устройства или строкой разрядов собственности, связанной с устройством, так, чтобы устройство принимало некоторый материал локального ключа, связанный с IEEE адресом, во время производства.

Получение общего ключа может использоваться для многих применений. Обычно, общий ключ будет являться криптографическим симметричным ключом. Симметричный ключ может использоваться для конфиденциальности, например исходящие или входящие сообщения могут быть зашифрованы с помощью симметричного ключа. Только устройство с доступом к обоим идентификационным номерам и к одному из двух материалов локального ключа (или доступом к материалу корневого ключа) будет способно расшифровать сообщения. Симметричный ключ может использоваться для аутентификации, например исходящие или входящие сообщения могут быть аутентифицированы с помощью симметричного ключа. Таким образом может быть подтверждено происхождение сообщения. Только устройство с доступом к обоим идентификационным номерам и к одному из двух материалов локального ключа (или доступом к материалу корневого ключа) будет способно создать аутентифицированные сообщения.

Способ конфигурирования сетевого устройства для использования общего ключа обычно будет выполняться посредством сетевой службы, например доверенной третьей стороны. Сетевая служба может получать необходимый материал, например материал корневого ключа, из другого источника, но также может генерировать его сама. Например, публичный модуль может быть сгенерирован. Например, частный модуль может быть сгенерирован, даже если публичный модуль является системным параметром и был принят.

В варианте осуществления один или более или все публичные модули N выбираются так, чтобы они удовлетворяли $2^{(a+2)b-1} \leq N \leq 2^{(a+2)b}-1$, где a представляет степень двумерного полинома, а b представляет длину ключа. Например, в варианте осуществления $N=2^{(a+2)b}-1$. Операция по модулю для последнего выбора может быть реализована особенно эффективно.

Наличие фиксированного публичного модуля имеет преимущество, состоящее в том, что его не нужно передавать на сетевые устройства, но можно объединить, например, с их системным программным обеспечением. В частности, публичный модуль может быть выбран, используя генератор случайных чисел.

Общественные и частные модули могут быть представлены в виде строки разрядов. Их также можно сократить, используя каждую конкретную математическую конструкцию. Например, вместо сохранения частного модуля, можно также сохранять его разницу с публичным модулем, которая намного короче.

Наличие частного модуля, выбранного таким образом, что все из ‘имеющего длину ключа’ количества самых младших разрядов бинарного представления разницы между публичным модулем и частным модулем являются нулевыми разрядами, увеличивает вероятность того, что общий ключ на первом сетевом устройстве из пары сетевых устройств будет близок к общему ключу, полученному на втором сетевом устройстве из пары сетевых устройств; то есть бинарное представление частного модуля содержит такие же разряды в ‘имеющих длину ключа’ самых младших позициях, как и бинарное представление публичного модуля. Например, если длина ключа составляет 64, частный модуль может быть выбран посредством вычитания числа, кратного 2^64, из публичного модуля. В варианте осуществления разница публичного модуля и частного модуля, поделенная на двойку в степени длины ключа, меньше чем двойка в степени длины ключа.

В варианте осуществления множество частных модулей получаются или генерируются в электронной форме, для каждого частного модуля из множества частных модулей выбирается симметричный двумерный полином, имеющий целые коэффициенты, чтобы получить множество симметричных двумерных полиномов, так, чтобы каждому частному модулю соответствовал симметричный двумерный полином. Определение одномерного полинома содержит подстановку идентификационного номера в каждый из множества симметричных двумерных полиномов, приведение по модулю частного модуля из множества частных модулей, соответствующих одному симметричному двумерному полиному, и сложение вместе множества результатов множества приведений. Наличие множества симметричных двумерных полиномов для разных модулей увеличивает безопасность, так как несовместимые конструкции дополнительно смешиваются. Обычно частные модули различаются. Наличие множества частных модулей дополнительно усложняет анализ, если соответствующие алгебраические конструкции являются очень разными; например выбирая их взаимно простыми, более конкретно, попарно взаимно простыми, еще более конкретно, выбирая их разными простыми числами.

Наличие разных частных модулей, и, в частности, множества частных модулей, усложнит анализ для взломщика. Чтобы еще больше увеличить безопасность, возможны дополнительные воздействия на коэффициенты. В варианте осуществления служба, суммирующая множество результирующих одномерных полиномов множества приведений, проверяет, не является ли значение каждого из результирующих коэффициентов либо слишком маленьким, либо слишком большим, например меньше чем минимальное пороговое значение, или больше чем максимальное пороговое

значение. Это еще больше увеличивает безопасность, так как в любом из двух случаев взломщик может выявить компоненты множества приведений, если они слишком большие или слишком маленькие. Например, если значение коэффициента, получившееся после сложения, равняется 1, и имеется только два одномерных полинома, тогда взломщик знает, что либо соответствующий коэффициент, связанный с первым полиномом, равен 1, а коэффициент, связанный со вторым полиномом, равен 0, либо наоборот. В частности, служба, генерирующая материал локального ключа для устройства, может проверять, является ли значение каждого из результирующих коэффициентов материала локального ключа по меньшей мере 'минимальным значением' или самое большое 'максимальным значением'. Эта проверка может быть опущена, в частности, если публичный модуль относительно близок к всем частным модулям, и все элементы материала ключа находятся между 0 и N-1. Если TTP может назначать идентификационные номера, она также могла бы назначать устройству другой идентификационный номер, если TTP обнаруживает маленькие или большие коэффициенты.

В варианте осуществления каждый конкретный частный модуль является таким, что все самые младшие имеющие длину ключа (b) последовательные разряды бинарного представления разницы публичного модуля и частного модуля являются нулевыми разрядами.

Общественный модуль может быть как больше, так и меньше частного модуля. В варианте осуществления бинарное представление разницы публичного модуля и частного модуля содержит имеющие по меньшей мере длину ключа разряды, равные нулю. Нулевые разряды в имеющих по меньшей мере длину ключа нулевых разрядах являются последовательными и могут присутствовать в любом месте в бинарном представлении. Наличие строки из нулевых битов в разнице публичного модуля и частного модуля предотвращает ситуацию, в которой запутывание заходит слишком далеко. Отметим, что строка может, но не обязана присутствовать во всех наборах параметров.

В варианте осуществления имеется целочисленный параметр 's', такой, что все самые младшие имеющие длину ключа разряды разницы публичного модуля и частного модуля, деленной на двойку в степени s, являются нулевыми разрядами. Параметр 's' одинаков для всех частных модулей, но может различаться для разных наборов параметров.

Например, можно определить делитель строки нулевых разрядов, который является степенью двойки, такой, что каждый конкретный частный модуль является таким, что все имеющие длину ключа (b) разряды бинарного представления разницы публичного модуля и частного модуля, деленной на делитель строки нулевых разрядов, являются нулевыми разрядами. Если самые младшие разряды являются нулевыми, делитель строки нулевых разрядов может быть взят равным 1. В варианте осуществления делитель строки нулевых разрядов больше 1. Деление на степень двойки должно интерпретироваться, как целочисленное деление, дающее такой же результат, как смещение разрядов в направлении самых младших разрядов. Любой остаток от деления игнорируется.

Чтобы сгенерировать общий ключ из имеющих длину ключа разрядов, сетевые устройства сначала применяют дополнительный этап деления. Первое сетевое устройство оценивает материал ключа для идентификационного номера второго сетевого устройства по модулю публичного модуля для каждого набора параметров и суммирует результаты, затем делит на 2^s и приводит по модулю двойки в степени длины ключа.

Отметим, что это эквивалентно тому, чтобы сначала применить приведение по модулю 2^(s+длина ключа), затем приведение по публичному модулю, а после этого разделить на 2^s. Здесь, «деление» включает в себя округление вниз.

В варианте осуществления частный модуль генерируется, используя генератор случайных чисел. В варианте осуществления множество частных модулей генерируются так, чтобы они были попарно взаимно простыми. Например, множество частных модулей могут быть сгенерированы итеративно, проверяя, для каждого нового частного модуля, что они все еще попарно взаимно простые, и если нет, отбрасывая последний сгенерированный частный модуль. Вариант осуществления содержит итеративное генерирование потенциального модуля, используя генератор случайных чисел, так, что все имеющие длину ключа (b) последовательные разряды бинарного представления разницы публичного модуля и частного модуля являются нулевыми разрядами, например, самые младшие имеющие длину ключа последовательные разряды, до тех пор пока потенциальный модуль не удовлетворит тесту простоты, используя устройство тестирования простоты, при этом полученный таким образом потенциальный модуль, удовлетворивший тесту простоты, используется в качестве частного модуля. Тест простоты может, например, являться тестом простоты Миллера-Рабина или тестом простоты Соловея-Штрассена.

Симметричный двумерный полином в переменных x и y степени a содержит только одночлены вида x^iy^j, с i<a, j<a. Более того, коэффициент, соответствующий x^iy^j совпадает с коэффициентом, соответствующим x^jy^i. Это может использоваться, чтобы сократить количество сохраняемых коэффициентов примерно вдвое. Отметим, что используется более слабое определение степени. Мы определяем степень одночлена, как максимальную степень переменных в одночлене. Поэтому степень x^iy^j равняется max (i,j), где i<a, j<a. Поэтому, например, то, что мы называем полиномом степени 1, имеет общий вид a+bx+cy+dxy (отметим, что, так как рассматриваются только симметричные полиномы, b=c). Отметим, что при желании можно использовать дополнительные ограничения на двумерный полином, включая, например, что используются только одночлены с i+j<a, но это не обязательно.

В варианте осуществления симметричный двумерный полином генерируется посредством сетевой службы. Например, симметричный двумерный полином может являться случайным симметричным двумерным полиномом. Например, коэффициенты могут выбираться в виде случайных чисел, используя генератор случайных чисел.

Хотя используемое запутывание значительно увеличивает устойчивость к атаке, в частности, в борьбе против коллюзионных атак, в которых объединяется множество материалов локального ключа, оно имеет потенциальный недостаток. Иногда общий ключ, полученный первым сетевым устройством, не во всех разрядах идентичен общему ключу, полученному посредством второго сетевого устройства. Это главным образом происходит из-за несовпадений в разрядах переноса после прибавления запутывающих коэффициентов. Другая причина состоит в эффекте потерь модульных эффектов каждого из частных модулей во время генерирования ключа, который влияет на сгенерированные биты переноса. Несмотря на неудобство этот недостаток может быть разрешен разными путями. Посредством более внимательного выбора запутывания вероятность различий и, в частности, вероятность больших различий может быть значительно снижена. Более того, было обнаружено, что различия, если они вообще есть, вероятнее всего располагаются в самых младших разрядах сгенерированных ключей. Поэтому посредством удаления одного или более самых младших разрядов, вероятность

идентичного общего ключа может быть увеличена. Например, в варианте осуществления способ определения общего ключа содержит определение того, получили ли первое сетевое устройство и второе сетевое устройство одинаковый общий ключ или нет, и если нет, получение дополнительного общего ключа из результата приведения по модулю ключевого модуля. Далее, общие ключи могут получать до тех пор, пока не будет обнаружено, что они равны с обеих сторон. Если в общем ключе остается меньшее количество разрядов чем пороговое значение, способ может быть завершен. Для некоторых применений, можно просто принять, что некоторая доля сетевых устройств не может взаимодействовать. Например, в специальных беспроводных сетях, в которых сообщение может быть направлено вдоль разных путей, потери возможности соединения не происходит, если некоторые сетевые устройства не могут взаимодействовать друг с другом.

В варианте осуществления некоторое количество самых младших разрядов общего ключа, которые удаляются, например, количество удаляемых разрядов может составлять 1, 2 или более, 4 или более, 8 или более, 16 или более, 32 или более, 64 или более. Посредством удаления большего количества самых младших разрядов, шанс наличия ключей, которые не равны, уменьшается; в частности, их можно сокращать до любого желаемого порогового значения. Вероятность того, что общие ключи равны, может быть вычислена посредством следующих математических соотношений, и также может быть определена экспериментально.

Хотя выбор запутывающих чисел может контролироваться, в варианте осуществления диапазон, из которого выбирается запутывающее число, уменьшается для коэффициентов, соответствующих одночленам более высокой степени. В частности, можно потребовать, чтобы $|\epsilon_{i,k}^{A}| < 2^{(a+2-k)b-2}$, где $\epsilon_{A,i}$ обозначает запутывающее число для i-го одночлена, i обозначает степень одночлена, соответствующего коэффициенту, a представляет степень двумерного полинома, a b представляет длину ключа. A представляет сетевое устройство, для которого генерируется материал локального ключа. В варианте осуществления запутывающее число генерируется для каждого коэффициента, например, используя вышеприведенную формулу. Разное запутывание может применяться для разных сетевых устройств. Например, даже если имеется 3 или более сетевых устройств, для каждого сетевого устройства могут быть сгенерированы разные запутывающие числа.

Отметим, что запутывающее число может быть ограничено положительными числами, но это не обязательно, запутывающие числа могут быть отрицательными. В варианте осуществления запутывающие числа генерируются, используя генератор случайных чисел. Множество запутывающих чисел могут быть сгенерированы и прибавлены к коэффициентам одномерного полинома, чтобы получить запутанный одномерный полином. Один или более, предпочтительно даже все коэффициенты одномерного полинома могут быть запутаны таким образом.

Количество разрядов в идентификационном номере сетевого устройства обычно выбирается, как меньшее или равное длине ключа. Идентификационный номер может являться строкой разрядов, скажем из 32 или 64, или более разрядов. Длина ключа может составлять 32 или более, 48 или более, 64 или более, 96 или более, 128 или более, 256 или более. Длина ключа может быть выбрана некоторым количеством разрядов более высокого порядка, чтобы снизить соответствующее количество самых младших разрядов определенного общего ключа. С другой стороны, в варианте осуществления

длина идентификационного номера выше, чем длина ключа. В этом случае эффект модульных операций может привести к более сильному влиянию на самые младшие разряды имеющих длину ключа разрядов сгенерированного ключа, так, что эти разряды могут не равняться для пары устройств, желающих сгенерировать общий ключ. Наличие более длинного идентификатора может, однако иметь положительный эффект в смысле безопасности, так как большее количество разрядов смешиваются вместе при выполнении соответствующих вычислений.

Устройство манипулирования полиномом может быть реализовано в программном обеспечении, работающем на компьютере, скажем, на интегральной схеме. Устройство манипулирования полиномом может быть очень эффективно реализовано в аппаратном обеспечении. Также возможна их комбинация. Например, устройство манипулирования полиномом может быть реализовано посредством манипулирования массивами коэффициентов, представляющих полиномы.

Электронное сохранение сгенерированного материала локального ключа на сетевом устройстве может быть реализовано посредством электронной передачи сгенерированного материала локального ключа на сетевое устройство, например, используя проводное соединение или используя беспроводное соединение, и сохранения сгенерированного материала локального ключа на сетевом устройстве. Это может быть выполнено во время производства или установки, например во время тестирования, интегральной схемы в сетевом устройстве. Тестовое оборудование может содержать сетевую службу или может быть соединено с ней. Это также может произойти после успешного присоединения устройства к рабочей сети (например, после получения доступа к сети или самостоятельной загрузки). В частности, материал локального ключа может быть распределен, как часть рабочих сетевых параметров.

Получение материала локального ключа для первого сетевого устройства в электронной форме может быть выполнено посредством электронного приема материала локального ключа от системы для конфигурирования сетевого устройства для использования общего ключа, например устройства сетевой службы. Получение материала локального ключа может также быть выполнено посредством извлечения материала локального ключа из локального хранилища, например памяти, такой как флэш-память.

Получение идентификационного номера для второго сетевого устройства может быть выполнено посредством приема идентификационного номера от второго сетевого устройства, например напрямую от второго сетевого устройства, например посредством беспроводного приема от второго сетевого устройства.

Общественный модуль и ключевой модуль могут храниться в сетевом устройстве. Они также могут быть получены от сетевой службы. Они также могут неявно находиться в программном обеспечении сетевого устройства. Например, в варианте осуществления ключевой модуль является степенью двойки. Приведение по модулю такого ключевого модуля может быть выполнено посредством отбрасывания всех разрядов за исключением самых младших имеющих длину ключа разрядов. Сначала результат подстановки приводится по модулю публичного модуля, потом он дополнительно приводится по модулю ключевого модуля.

Хоть это и не обязательно, публичный модуль и ключевой модуль могут являться взаимно простыми. Это может быть достигнуто посредством наличия нечетного публичного модуля и ключевого модуля, являющегося степенью 2. В любом случае избегается ситуация, в которой модуль ключа является делителем публичного модуля, так как в этом случае приведение по модулю публичного модуля может быть опущено.

Способ для согласования ключа между двумя устройствами может использовать множество двумерных полиномов в качестве материала корневого ключа. Можно использовать способ для согласования ключа, используя x-согласование между x сторонами посредством использования x случайных полиномов в качестве материала корневого ключа. В этом расширении доверенная третья сторона оценивает x случайных полиномов в переменной в соответствующем кольце, результирующие x-1 случайных полиномов затем суммируются по целым числам, генерируя материал локального ключа, сохраняемый на устройстве. Когда x устройств должны согласовать ключ, устройство оценивает свой материал локального ключа в идентификаторах других x-1 устройств.

Использование ассиметричных двумерных полиномов в качестве материала корневого ключа, например f(x,y)!=f(y,x), позволяет приспособить создание двух групп устройств, например устройства в первой группе принимают KM(Id,y), а устройства во второй группе принимают KM(x,iD), где KM - материал локального ключа, сохраненный на устройстве. Два устройства, принадлежащих к одной группе, не могут генерировать общий ключ, но два устройства из разных групп могут. Смотрите дополнительно Blundo.

Идентификационный номер сетевого устройства может быть вычислен, как односторонняя функция строки разрядов, содержащей информацию, связанную с устройством. Односторонняя функция может являться криптографической хэш-функцией, такой как SHA2 или SHA3. Выход односторонней функции может быть урезан так, чтобы он соответствовал размеру идентификатора. В качестве альтернативы размер односторонней функции меньше чем максимальный размер идентификатора.

В варианте осуществления симметричные полиномы используют один многочлен вида $(ax^i y^i)_{p_j}$, где $<>_p$ представляет модульную операцию. В этом случае элементы находятся в конечной группе, а операция является умножением. Публичный модуль может быть больше, чем частный модуль, или меньше; если имеется множество частных модулей, некоторые из них могут быть больше, чем частный модуль, а некоторые могут быть меньше.

Материал корневого ключа может быть оценен на любом кольце. Возможно использовать полиномы в виде одного многочлена, такого как Ax^a, в случае чего может использоваться группа.

Аспект изобретения относится к системе для конфигурирования сетевого устройства для использования общего ключа, например, сетевой службе, система, содержащая узел получения материала ключа для получения в электронной форме по меньшей мере двух наборов параметров, набор параметров, содержащий частный модуль, публичный модуль и двумерный полином, имеющий целые коэффициенты, бинарное представление публичного модуля и бинарное представление частного модуля являются одинаковыми в имеющих по меньшей мере длину ключа последовательных разрядах, генератор для генерирования материала локального ключа для сетевого устройства, содержащий узел управления сетевым устройством для получения в электронной форме идентификационного номера для сетевого устройства, и для электронного сохранения сгенерированного материала локального ключа на сетевом устройстве, и устройство манипулирования полиномом, генератор, сконфигурированный, чтобы, для каждого набора параметров по меньшей мере двух наборов параметров, получать

соответствующий одномерный полином посредством: определения, используя устройство манипулирования полиномом, одномерного полинома из двумерного полинома из набора параметров посредством подстановки идентификационного номера в упомянутый двумерный полином, и приведения результата подстановки по модулю частного модуля набора параметров, и электронное сохранение на сетевом устройстве сгенерированного материала локального ключа, содержащего публичный модуль каждого набора параметров и соответствующий одномерный полином каждого набора параметров.

Вариант осуществления изобретения содержит генератор запутывающих чисел, например генератор случайных чисел, для генерирования для генерирования запутывающего числа, устройство манипулирования полиномом сконфигурировано, чтобы прибавлять запутывающее число к коэффициенту одномерного полинома, чтобы получить запутанный одномерный полином, сгенерированный материал локального ключа, содержащий запутанный одномерный полином. Запутывающее число может быть представлено в виде коэффициента запутывающего полинома. В варианте осуществления каждый коэффициент суммы запутывающих полиномов кратен 2 в степени длины ключа. В варианте осуществления каждый коэффициент суммы запутывающих полиномов, деленный на степень двойки, кратен 2 в степени длины ключа. Деление на степень двойки может вычисляться посредством округления вниз.

Аспект изобретения относится к первому сетевому устройству, сконфигурированному, чтобы определять общий ключ, являющийся криптографическим ключом, сетевое устройство, содержащее: узел получения материала локального ключа для получения материала локального ключа для первого сетевого устройства в электронной форме, материал локального ключа, содержащий по меньшей мере два, необязательно, запутанных, одномерных полинома и соответствующие публичные модули, приемник для получения идентификационного номера для второго сетевого устройства, второе сетевое устройство, отличное от первого сетевого устройства, устройство манипулирования полиномом, сконфигурированное, чтобы выполнять, для каждого из по меньшей мере двух, необязательно, запутанных, одномерных полиномов: подстановку идентификационного номера второго сетевого устройства в упомянутый одномерный полином, и приведение результата подстановки по модулю публичного модуля, соответствующего упомянутому одномерному полиному, и сложение вместе результатов приведений по модулю публичного модуля и приведение по модулю ключевого модуля, и устройство получения ключа для получения общего ключа из результата приведения по модулю ключевого модуля.

Устройство получения ключа может быть реализовано в виде компьютера, например интегральной схемы, выполняющей программное обеспечение, в виде аппаратного обеспечения, в виде их комбинации, и подобного, и оно сконфигурировано для получения общего ключа из результата приведения по модулю ключевого модуля.

Получение общего ключа из результата приведения по модулю ключевого модуля может включать в себя применение функции получения ключа, например функции KDF, определенной в спецификации Открытого мобильного альянса (Open Mobile Alliance) OMA DRM (OMA-TS-DRM-DRM-V2 0 2-20080723-A, секция 7.1.2 KDF), и схожих функций. Получение общего ключа может включать в себя отбрасывание одного или более самых младших разрядов (перед применением функции получения ключа). Получение общего ключа может включать в себя прибавление, вычитание или конкатенацию целого числа (перед применением функции получения ключа).

Множество сетевых устройств, каждое из которых имеет идентификационный номер

и соответствующий материал локального ключа, вместе могут формировать сеть связи, для безопасного, например конфиденциального и/или аутентифицированного взаимодействия между парами сетевых устройств.

Генерирование ключа основано на ID (идентификаторе) и позволяет генерировать попарные ключи между парами устройств. Первое устройство A может опираться на алгоритм, который получает ключ из материала локального ключа и идентификационного номера.

В варианте осуществления первое сетевое устройство передает сообщение подтверждения ключа на второе сетевое устройство. Например, сообщение подтверждения может содержать шифровку сообщения, и, необязательно, само сообщение. Второе сетевое устройство может проверять шифровку сообщения. Сообщение может быть фиксированным и присутствовать на втором устройстве, чтобы избежать необходимости его передачи. Сообщение может являться случайным, или одноразовым, и т. д., в случае чего оно может передаваться вместе с шифровкой. Второе устройство может ответить с помощью сообщения, которое содержит индикацию того, согласуются ли ключи. Второе устройство может также ответить с помощью собственного сообщения подтверждения ключа. Если первое и/или второе устройство обнаружит, что ключи не равны, они могут начать процесс уравнивания ключа, например, посредством удаления самых младших разрядов, и т. д.

Сетевые устройства и система могут являться электронными устройствами. Сетевые устройства могут являться мобильными сетевыми устройствами.

Способ согласно изобретению может быть реализован на компьютере в виде машинно-реализованного способа, или в специальном аппаратном обеспечении, или в комбинации обоих. Исполняемый код для способа согласно изобретению может храниться в компьютерном программном продукте. Примеры компьютерных программных продуктов включают в себя устройства памяти, оптические устройства хранения, интегральные схемы, сервера, онлайн программное обеспечение, и т.д. Предпочтительно компьютерный программный продукт содержит энергонезависимое средство программного кода, хранящееся на машиночитаемом носителе, для выполнения способа согласно изобретению, когда программный продукт выполняется на компьютере.

В предпочтительном варианте осуществления компьютерная программа содержит средство компьютерного программного кода, приспособленное, чтобы выполнять все этапы способа согласно изобретению, когда компьютерная программа запускается на компьютере. Предпочтительно компьютерная программа может быть реализована на машиночитаемом носителе.

КРАТКОЕ ОПИСАНИЕ ЧЕРТЕЖЕЙ

Эти и другие аспекты изобретения станут очевидны из, и будут разъяснены со ссылкой на варианты осуществления описанные далее в материалах настоящей заявки. На чертежах,

Фигура 1 - блок-схема, иллюстрирующая генератор материала корневого ключа,

Фигура 2 - блок-схема, иллюстрирующая генератор материала локального ключа,

Фигура 3 - блок-схема, иллюстрирующая сеть связи,

Фигура 4 - блок-схема последовательности операций, иллюстрирующая генерирование материала локального ключа,

Фигура 5 - блок-схема последовательности операций, иллюстрирующая генерирование общего ключа,

Фигура 6 - схема последовательности операций, иллюстрирующая генерирование

общего ключа.

Стоит отметить, что элементы, которые имеют одинаковые номера ссылок на разных фигурах, имеют одинаковые структурные признаки и одинаковые функции, или являются одинаковыми сигналами. Если функция и/или структура такого элемента уже была объяснена, нет необходимости повторять это объяснение в подробном описании.

ПОДРОБНЫЕ ВАРИАНТЫ ОСУЩЕСТВЛЕНИЯ

Несмотря на то что это изобретение допускает варианты осуществления в множестве разных форм, один или более конкретных вариантов осуществления показаны на чертежах и будут подробно описаны в материалах настоящей заявки с пониманием того, что настоящее раскрытие должно рассматриваться в качестве примера принципов изобретения, и не предназначено для ограничения изобретения показанными и описанными конкретными вариантами осуществления.

Далее описан вариант осуществления способа использования общего ключа. Способ содержит фазу установления и фазу использования. Фаза установления может включать в себя этапы инициации и этапы регистрации. Этапы инициации не вовлекают сетевые устройства.

Этапы инициации выбирают параметры системы. Этапы инициации могут выполняться доверенной третьей стороной (TTP). Тем не менее, параметры системы также могут рассматриваться, как задаваемые в виде входных данных. В этом случае доверенная третья сторона не должна их генерировать, и этапы инициации могут быть пропущены. Например, доверенная третья сторона может принимать параметры системы от производителя устройства. Производитель мог выполнить этапы инициации, чтобы получить параметры системы. Для удобства описания, мы будем указывать ссылкой на доверенную третью сторону, как на выполняющую этапы инициации, не забывая, что это не является обязательным.

На этапах инициации устанавливается несколько наборов параметров. При наличии идентификационного номера сетевого устройства, наборы параметров используются, чтобы генерировать материал локального ключа; из каждого набора параметров получаются одномерные полиномы и соответствующий публичный модуль. Сетевому устройству предоставляется материал локального ключа, но не предоставляется доступ к наборам параметров. Так как наборы параметров позволяют сгенерировать новый материал локального ключа, они известны только доверенной стороне и держатся в секрете от обычных сетевых устройств.

Сетевое устройство A может генерировать общий ключ из своего материала локального ключа и идентификационного номера другого устройства B. Чтобы сделать это, сетевое устройство A выполняет вычисления, используя свой материал локального ключа.

Этапы инициации

На этапах инициации выбирается материал корневого ключа. Несколько параметров являются глобальными параметрами.

Выбирается требуемая длина ключа для ключа, который будет общим для устройств на фазе использования; эта длина ключа указана ссылкой, как ‘b’. Обычное значение для применения с низкой безопасностью может составлять от 64 до 80. Обычное значение для потребительского уровня безопасности может составлять 128. Применения с высокой секретностью могут предпочитать 256 или даже более высокие значения. Может не существовать прямой зависимости между надежностью безопасности алгоритма и b; обеспечиваемая безопасность будет составлять не более b. В зависимости от будущих алгоритмов для атаки системы безопасность алгоритма может быть меньше чем b.

Выбирается количество наборов параметров, которые будут сгенерированы; количество наборов параметров указано ссылкой, как 't'. Высокое значение t предполагает, что атаковать результирующую систему, например, используя методики на основе решеток, сложнее. С другой стороны, более высокое значение t также предполагает более высокие вычислительные требования и требования памяти на сетевых устройствах. Для применений с очень низкой безопасностью, возможно значение t=1, однако это может подразумевать, что при наличии достаточного количества компрометированных ключей лежащий в основе материал ключа может быть восстановлен. Рекомендуется использовать по меньшей мере значение t=2; это значение уже вызывает существенное увеличение сложности требуемого криптоанализа, например, атак на основе решеток. Тем не менее, для применений с более высокой безопасностью, может использоваться значение 3, 4 или даже выше.

Затем выбираются количества t наборов параметров. Каждый набор параметров j, где j=1,..., t, содержит требуемую степень a, публичный модуль N, по меньшей мере один частный модуль p_1, и по меньшей мере один симметричный двумерный полином f_1. Когда это будет удобно, публичный индекс будет обозначаться индексом, чтобы указывать на набор параметров, к которому он принадлежит: N_j.

Преимущественные пути выбора этих параметров обсуждаются ниже. В частности, двумерные полиномы каждого набора параметров являются чувствительными к безопасности, и не будут раскрываться для обычных сетевых устройств; также нет причин раскрывать частные модули, поэтому рекомендуется держать их в секрете, знание о них может даже облегчить атаку на систему. Длина ключа b и публичный модуль N_j нужны на сетевом устройстве и не могут держаться в секрете от доверенной стороны.

Каждый набор параметров вносит вклад в сложность лежащей в основе сложной задачи. Как будет пояснено ниже, некоторые выборы параметров будут вызывать более сложную задачу, чем другие выборы. В принципе, выбор наборов параметров является независимым, например можно выбрать один набор параметров со значениями, соответствующими более высокой безопасности, и выбрать второй набор с меньшими параметрами. В этом случае второй и/или дальнейшие наборы главным образом вносят вклад в избегание атак на сложный набор. В этой ситуации может быть в некотором смысле проще получить ограничения на безопасность. С другой стороны, можно также выбрать все наборы параметров сопоставимой сложности. В последней ситуации сложность задачи формируется всеми наборами. Это оптимизирует вычислительные ресурсы на сетевом устройстве.

Этапы выбора набора параметров

Эти этапы будут повторяться t раз; по одному разу для каждого требуемого набора параметров.

Выбирается требуемая степень; степень управляет степенью определенных полиномов. Степень будет указываться ссылкой, как 'a', она составляет по меньшей мере 1. Практичным выбором для a является 2. Более безопасное применение может использовать более высокое значение a, например 3 или 4, или даже выше. Для простого применения возможно a=1. Случай a=1 относится к так называемой 'задаче скрытого числа'; более высокие значения 'a' относятся к задаче шумной полиномиальной интерполяции, подтверждая, что эти случаи являются сложными для взлома.

Выбирается количество полиномов. Количество полиномов будет указано ссылкой как 'm'. Практичным выбором для m является 2. Более безопасное применение может

использовать более высокое значение m, например 3 или 4, или даже выше. Отметим, что применение с низкой сложностью может использовать низкое значение m, так как высокое значение m подразумевает более высокую сложность реализации на TTP.

Более высокие значения параметров безопасности a и m увеличивает сложность системы и, соответственно, увеличивает ее защищенность. Более сложные системы сложнее анализировать, и, следовательно, они более устойчивы к криптоанализу. Степень a может для удобства быть одинаковой для всех наборов параметров, кроме того, m тоже может быть одинаковым для всех наборов параметров; отметим, что это не является необходимым.

В варианте осуществления публичный модуль N выбирается удовлетворяющим условию $2^{(a+2)b-1} \leq N$, и наиболее предпочтительно - также условию $N \leq 2^{(a+2)b}-1$. Границы не являются строго обязательными; система также могла бы использовать меньшее/большее значение N, хотя это не считается наилучшим вариантом.

Часто длина ключа, степень и количество полиномов будут являться предопределенными, например, разработчиком системы и будут предоставляться доверенной стороне в виде входных данных. В качестве практичного выбора можно взять $N=2^{(a+2)b}-1$. Например, если a=1, b=64, тогда N может равняться $N=2^{192}-1$. Например, если a=2, b=128, тогда N может равняться $N=2^{512}-1$. Выбор для N верхней или нижней границы из вышеупомянутого интервала имеет преимущество легкого вычисления. Чтобы увеличить сложность для взломщика, можно выбрать случайное число в пределах диапазона для N.

Множество из m частных модулей $p_1, p_2, \ldots, p_m$ выбирается доверенной третьей стороной (TTP). Модули являются положительными целыми числами. Во время этапов регистрации каждое устройство будет связано с идентификационным номером. Каждый выбранный частный модуль больше, чем самый большой используемый идентификационный номер. Например, можно ограничить идентификационные номера посредством требования, чтобы они были меньше или равнялись 2^b-1, и чтобы выбранные частные модули были больше чем 2^b-1. Каждое выбранное число удовлетворяет следующему соотношению $p_j=N+\gamma_j \cdot 2^b$. При этом γ_j - это целые числа, такие, что $|\gamma_j|<2^b$. Одним практичным способом выбора чисел, которые удовлетворяют этому требованию, является выбор m случайных целых чисел γ_j, таких, что $-2^b+1<\gamma_j<2^b-1$, и расчет выбранных частных модулей из соотношения $p_j=N+\gamma_j \cdot 2^b$. Наличие немного большего $|\gamma_j|$ может допускаться, однако может возникнуть проблема, в которой модульная операция зайдет слишком далеко, так, что общие ключи могут не быть равными.

Для m>1, система является более сложной, и, таким образом, более надежной, так как операции деления по модулю для разных модулей объединяются, хоти такие операции и не являются совместимыми в обычном математическом смысле. По этой причине лучше выбирать отобранные частные модулю попарно различными.

Генерируется множество m симметричных двумерных полиномов $f_1, f_2, \ldots, f_m$ степеней a_j. Все степени удовлетворяют $a_j \leq a$, $a=MAX\{a_1, \ldots, a_m\}$. Практичным выбором является взять каждый полином степени a. Двумерный полином является полиномом двух переменных. Симметричный полином f удовлетворяет f(x,y)=f(y,x). Каждый полином

f_j оценивается в конечном кольце, сформированном целыми числами по модулю p_j, полученных посредством приведения по модулю p_j. Целые числа по модулю p_j формируют конечное кольцо с p_j элементами. В варианте осуществления полином f_j представлен коэффициентами от 0 до p_j. Двумерные полиномы могут выбираться случайно, например посредством выбора случайных коэффициентов в этих границах.

Безопасность использования общего ключа зависит от этих двумерных полиномов, так как они являются материалам корневого ключа системы; поэтому предпочтительно строгие меры принимаются, чтобы защитить их, например процедуры управления, устойчивые к взлому устройства, и подобное. Предпочтительно выбранные целые числа $p_1, p_2, \ldots, p_m$ также держаться в секрете, включая значение γ_j, соответствующее p_j, хотя это менее критично. Будем указывать ссылкой на двумерные полиномы также в следующей форме: для j=1, 2, …, m будем писать $f_j(x,y) = \sum_{i=0}^{a} f_{i,j}(x)y^i$.

Вышеприведенный вариант осуществления может быть изменен множеством путей. Ограничения на публичные и частные модули могут выбираться множеством разных путей, так, чтобы было возможным запутывание одномерного полинома, и чтобы в то же время общие ключи, получаемые на сетевых устройствах, оставались достаточно близкими друг к другу достаточно часто. Как поясняется, то, что считается достаточным, будет зависеть от применения, требуемого уровня безопасности и вычислительных ресурсов, доступных на сетевых устройствах. Вышеупомянутый вариант осуществления объединяет положительные целые числа так, что модульные операции, которые выполняются при генерировании долей полиномов, объединяются нелинейным образом, когда они суммируются по целым числам, создавая нелинейную структуру для материала локального ключа, хранящегося на сетевом устройстве. Вышеприведенный выбор N и p_j имеет следующее свойство: (i) размер N фиксирован для всех сетевых устройств и связан с a; (ii) нелинейный эффект появляется на самых старших разрядах коэффициентов, формирующих материал ключа, хранящийся на устройстве. Из-за этой конкретной формы общий ключ может быть сгенерирован посредством сокращения модуля 2^b после приведения по модулю N.

Эти конструктивные принципы могут применяться более общим образом, чтобы улучшить аспекты (i) и (ii), как упоминалось в последнем абзаце. Ниже приведены различные общие конструкции для выбора публичных и частных модулей. Чтобы обратиться к первому пункту (i), эта конструкция для N и p_j удовлетворяет более общему выражению, в котором запишем $p_j=2^X+\gamma_j 2^{Y_j}-1$ так, что для каждого j, $Y_j+ba_j=X$ и $|\gamma_j|<2^b$. Это выражение обеспечивает более изменяемую форму p_j, в то же время гарантируя максимальный эффект при введении нелинейных эффектов. Отметим, что можно также сделать, чтобы $Y_j+ba_j\approx X$, где разница между левой и правой стороной является долей длины ключа.

Чтобы обратиться ко второму пункту, вышеприведенная форма для N и p_j удовлетворяет еще более общему выражению, в котором $p_j=\beta 2^X+\gamma_j 2^{Y_j}+\zeta_j 2^{Z_j}$. Посредством установки, например, $\beta=1$, $\zeta_j=-1$ $Z_j=0$ $\forall j$, мы получаем предыдущее выражение, в котором разные значения γ_j вводят нелинейный эффект в самых старших разрядах коэффициентов материала ключа, хранящегося на сетевом устройстве. В этом случае постоянный публичный модуль (N) составляет $N=2^X-1$, в то время как частная переменная часть,

используемая при генерации разных положительных целых чисел, используемых в модульных операциях, составляет $\gamma_j 2^{Yj}$. В качестве альтернативы мы можем установить $\gamma_j=1$, $\beta=1$, $Y_j=(a_j+1)b$, $X=(a_j+2)b$ $Z_j=0$ $\forall j$, в то время как ζ_j разные для разных j, так, что $|\zeta_j|<2^b$. В этом случае различия в ζ_j позволяют вводить нелинейный эффект в самых младших разрядах коэффициентов материала локального ключа, хранящегося на узле. Конструкция публичной части в этом случае также отличается и равняется $N=\beta_j 2^{Xj}+\gamma_j 2^{Yj}=2^X+2^{(aj+1)b}$, то есть части, которая остается постоянной. Отметим, что в этом случае нелинейный эффект находится в самой младшей части, и, из-за условия на максимальный эффект смешивания, упомянутый выше, разница $Y_j-Z_j-\log_2(\zeta_j)$ должна составлять a_jb. Подобным образом другие конструкции могут быть определены, следуя той же концепции.

Как показано выше, возможны многие выборы параметров. Тем не менее, некоторые выборы будут давать лучшие реализации. Особенно важен выбор публичного модуля. Например, некоторые выборы публичных модулей обеспечивают эффективные операции по модулю. Кроме того, влияние публичного модуля на разряды, из которых мы берем ключ, скажем, самые младшие разряды (LSB), предпочтительно отличается. Отличающееся влияние может проверяться посредством выполнения операций для генерирования общего ключа и проверки, приводят ли различия в p_j к другому способу генерирования ключа. Это показано в примере, приведенном ниже:

Например, выгодно выбирать публичные модули, которые имеют небольшие различия в числе длины ключа самых младших разрядов, скажем, меньше чем предопределенные различия. Например, вариант осуществления может использовать такие числа, как t=2, и $N_1=2^{(a+2)b}-1$ и $N_2=2^{(a+2)b}-2-1$. В этом конкретном случае член -2 играет важную роль во время фазы генерирования ключа, так как приведенный модуль N_1 не будет включать этот эффект, а приведенный модуль N_2 будет включать его. Отметим, что приведение в этом случае относится к смещению разрядов переполнения, которые выше разрядов (a+2)6, в самую нижнюю часть.

Однако проблема с выбором публичных модулей, таким образом, состоит в том, что доступно только ограниченное число вариантов, которые меньше 2^b. В целом, мы хотим ввести член 2^h в N, такой, что h<b и h>1. Кроме того, проблема состоит в том, число разрядов, которые реально подвержены влиянию другой формы N, будет составлять примерно b-h, и будет всего около 2^h разных чисел. Чтобы преодолеть эти проблемы, например иметь больше вариантов для N и максимизировать число разрядов, которые могут использоваться для ключа, подверженных влиянию разных операций, можно использовать более общее определение p_i образом, представленным на два абзаца выше. В этом случае нелинейный эффект вводится как в самый старший разряд (MSB), так и в LSB полиномиальных коэффициентов, например, посредством использования $p_i=N-\gamma_i 2^{b(a+1)}-\zeta_i$ с соответствующим публичным модулем $N=2^{(a+2)b}-2^{ba}$. Как определено здесь, γ_i и ζ_i выбираются разными для разных p_i, предпочтительно также разными по всем наборам параметров. В этом случае ключ генерируется из средних разрядов, а не из LSB.

Схожим выбором является следующее. В этих уравнениях, первый индекс i обозначает набор параметров и принимает значения до t; второй индекс j обозначает количество

чисел p, используемое в наборе параметров, и принимает значения до m.

$$N_i = 2^{(a+2)b} - 2^{ba} - \zeta_i$$

$$p_{i,j} = N_i - \gamma_{i,j} 2^{b(a+1)}$$

Практичным выбором является взять t=2. Затем может быть выбран каждый набор параметров. Практичным выбором является взять m=2 для каждого набора параметров. В частности, в представленных непосредственно выше уравнениях, такой выбор является удачным. С такой конструкцией можно найти много N_i посредством изменения b-разрядных значений ζ, например, случайно. В этой конструкции параметры $\gamma_{i,j}$ выполняют смешивание при генерировании долей материала ключа, хранящихся на устройствах. Это может быть выполнено доверенной стороной. Параметры ζ выполняют смешивание ключей на устройстве. Наиболее предпочтительно в этом случае также добавляется шум, следуя той же мотивации, как и в варианте осуществления. В этом случае условие на шум должно быть обновлено так, чтобы сумма шума, то есть запутывающих полиномов, равнялась нулю в позиции, из которой извлекается ключ (например, из средних разрядов).

Предпочтительно, чтобы публичные модули, скажем, N_1 и N_2, не все были кратны 2^b. Это так, потому что для положительных целых чисел a, m, n и для подходящего целого числа q имеем, что $a=qmn+\langle a \rangle_{mn}$, и поэтому $a=\langle a \rangle_{mn} \bmod n$, откуда мы заключаем, что $\langle a \rangle_n = \langle \langle a \rangle_{mn} \rangle_n$. Как следствие, если и N_1 и N_2 кратно 2^b, тогда $\langle \langle F_\eta^1(\eta')\rangle_{N_1} + \langle F_\eta^2(\eta')\rangle_{N_2} \rangle_{2^b} = \langle F_\eta^1(\eta') + F_\eta^2(\eta')\rangle_{2^b}$. То есть задача сводится к случаю t=1.

Этапы регистрации

На этапе регистрации каждому сетевому устройству назначается материал ключа (KM). Сетевое устройство связано с идентификационным номером. Идентификационный номер может назначаться по требованию, например, посредством TTP или может уже храниться в устройстве, например сохраняться в устройстве при производстве, и т. д.

TTP генерирует набор материала ключа KM^A для устройства с идентификационным номером A посредством расчета t полиномов следующим образом:

$$F_i^A(X) = \sum_{j=1}^{m} < f_j(x,A) >_{p_j} + \sum_{k=0}^{a} \epsilon_{i,k}^A X^k = \sum_k C_{i,k}^A X^k$$

В этом уравнении индекс i обозначает наборы параметров, то есть принимает значения от 1 до t. Индекс j обозначает количество полиномов и частных модулей на набор параметров. Индекс k обозначает коэффициенты в запутывающем полиноме. Отметим, что выбирается по одному запутывающему полиному на набор параметров. Некоторые или все наборы параметров могут не содержать запутывающий полином. Кроме того, публичные модули из набора параметров, соответствующие вышеупомянутым одномерным полиномам, включены в материал локального ключа.

X является формальной переменой. Отметим, что материал ключа является нелинейным. Запись $< \cdots >_{p_j}$ обозначает приведение по модулю p_j каждого коэффициента полинома между скобками. Запись $\epsilon_{A,i}$ обозначает случайное целое число, которое является примером запутывающего элемента, такое,

что $|\epsilon_{i,k}^{A}| < 2^{(a+2-k)b-2}$. Отметим, что любое из случайных целых чисел может быть положительным или отрицательным. Случайные числа $\in$ генерируются заново для каждого устройства. Член $\sum_{k=0}^{a} \epsilon_{i,k}^{A} X^k$, таким образом, представляет для каждого i полином в X степени a, длина коэффициентов которого становится короче с увеличением степени. В качестве альтернативы более общее, но более сложное условие состоит в том, что $\sum_{k=0}^{a} |\epsilon_{i,k}^{A}| \cdot 2^{b+k}$ мала, например < 2a.

Все другие добавления могут либо использовать натуральную целочисленную арифметику, либо (предпочтительно) использовать сложение по модулю N_i. Таким образом, оценивание одномерных полиномов $\sum_{j=1}^{m} < f_j(x, A) >_{p_j}$ выполняется по отдельности по меньшему модулю p_j, но суммирование самих этих приведенных одномерных полиномов предпочтительно производится по модулю N. Кроме того, добавление запутывающего полинома $\sum_{k=0}^{a} \epsilon_{i,k}^{A} X^k$ может выполняться, используя традиционную целочисленную арифметику, или предпочтительно по модулю N.

Материал ключа содержит коэффициенты $C_{i,k}^{A}$ с k=0, …, a и i=1, …, t. Материал ключа может быть представлен в виде набора полиномов, как показано выше. На практике материал ключа может храниться в виде списка, например, двумерного массива целых чисел $C_{i,k}^{A}$. Устройство A также принимает числа N_i и b. Манипулирование полиномами может быть реализовано, например, в виде манипулирования массивами, содержащими коэффициенты, например, располагая все коэффициенты в списке в предопределенном порядке. Отметим, что полиномы могут быть реализованы в виде других структур данных, например в виде ассоциативного массива (также известного, как ‘карта’), содержащего коллекцию пар (степень, коэффициент), предпочтительно такую, что каждый коэффициент чаще всего появляется в коллекции один раз.

Коэффициенты C_i^A, которые предоставляются устройству, предпочтительно находятся в диапазоне 0, 1, …, N-1. Из-за небольшого размера идентификатора может случиться так, что не все разряды коэффициентов будут использоваться для генерирования ключа. В этом случае должны храниться только релевантные части коэффициентов.

Как указано в начале этого документа, чтобы уменьшить вероятность того, что устройство A получит общий ключ, отличный от его дополняющего устройства B, запутывающие полиномы могут выбираться так, чтобы для каждого k=0, …, a

$$\sum_{i=1}^{t} \varepsilon_{i,k} \equiv 0 \bmod 2^b$$

То есть сумма всех запутывающих полиномов кратна 2^b. Как уже упоминалось, это имеет хорошее свойство, согласно которому, если взломщик захочет удалить шум посредством добавления материалов ключа, тогда он будет смешивать материалы

ключа, полученные посредством выполнения модульных операций с другими модулями. Если он не добавляет их, тогда материалы ключа будут скрыты посредством шума.

В случае когда используется более общая конструкция для N и целые числа p_j, запутывающий полином должен быть приспособлен так, чтобы случайные числа $\in$ влияли на разные части коэффициентов. Например, если нелинейный эффект вводится в самые младшие разряды коэффициентов материала ключа, хранящегося на сетевых устройствах, тогда случайные числа должны влиять только на высшую часть коэффициентов и переменное число разрядов в низшей части коэффициентов. Это прямое расширение способа, описанного выше, но возможны и другие расширения.

Фаза использования

Когда два устройства A и B имеют идентификационный номер и уже получили свой материал ключа от TTP, они могут использовать свой материал ключа, чтобы получить общий ключ. Устройство A может выполнять следующие этапы, чтобы получить свой общий ключ. Сначала, устройство A получает идентификационный номер B устройства B, затем A генерирует общий ключ посредством расчета следующего:

$$K_{AB} = < \Sigma_i \ < F_i^A(X)|_{X=B} >_{N_i} >_{2^b} = < \Sigma_i \ < \Sigma_k \ C_{i,k}^A B^k >_{N_i} >_{2^b}$$

То есть A оценивает каждый из своих одномерных полиномов F_i^A из своего материала ключа для значения B; результат оценивания материала ключа является целым числом. Затем, устройство A приводит результат оценки первого модуля по соответствующему публичному модулю N_i. Далее, результаты оценивания всех полиномов F_i^A после модульного оценивания суммируется в виде целых чисел, и затем результат этого суммирования приводится по модулю ключа 2^b. Результат будет указываться ссылкой, как общий ключ устройства A, он является целым числом в диапазоне от 0 до 2^b-1. Для этой части устройство B может генерировать общий ключ устройства B посредством оценивания своего материала ключа для идентификатора A и приведения результата по модулю N, а затем по модулю 2^b, схожим образом с действиями устройства A.

В соответствии с вышеприведенным описанием, если используется более общее выражение для N и положительных целых чисел p_j, тогда способ получения b-разрядного ключа нуждается в небольшом приспособлении. В частности, мы можем взять $p_{i*m+j} = \beta 2^X + \gamma_{i*m+j} 2^{Y_{i,j}} + \delta 2^W + \zeta_i 2^{Z_i}$ для частных модулей и $N_i = \beta 2^X + \delta 2^W + \zeta_i 2^{Z_i}$ для публичных модулей, тогда это позволит ввести нелинейность в доли материала ключа посредством b-разрядного члена γ_{i*m+j}. Отметим, что в этой конкретной конструкции мы имеем m полиномов в каждом наборе материалы ключа, и каждый из этих полиномов проиндексирован идентификатором j. Более того, мы можем иметь вплоть до t разных наборов материала ключа, индексированных посредством i. Отметим также, что обычно $Y_{i,j}$ постоянно $\forall i,j$. Дополнительно, b-разрядные члены ζ_i различны для разных N_i с i=1, …, t, и являются теми, которые вводят нелинейный эффект при смешивании ключей, сгенерированных из разных материалов ключа на узле. В этом

случае ключ генерируется следующим образом:

$$K_{AB} = < \frac{\sum_i < F_i^A(X)|_{X=B} >_{N_i}}{2^W} >_{2^b}$$

Как видно, каждая из t долей материала ключа оценивается в x=B и приведены по модулю N_i. При этом приведении вводится эффект ζ_i. Так как наименьшая степень двойки, общая для всех p_{i*m+j}, равняется w, результат делится на 2^w, чтобы мог быть сгенерирован общий ключ.

Так как двумерные полиномы в материале корневого ключа симметричны, общий ключ устройства A и общий ключ устройства B часто, хоть и не обязательно всегда, равны. Конкретные требования на частные модули, целые числа $p_1, p_2, \ldots, p_m$ в наборах параметров и на случайные числа $\in$ таковы, что ключи часто равняются и почти всегда близки к друг другу по модулю двойки в степени длины ключа. Если A и B получили одинаковый общий ключ, они могут использовать его в качестве симметричного ключа, который разделен между A и B; например, он может использоваться во множестве криптографических применений, например, они могут обмениваться одним или более сообщениями, зашифрованными и/или аутентифицированными, используя общий ключ. Предпочтительно алгоритм получения ключа применяется к общему ключу для дополнительной защиты главного ключа, например, может применяться хэш-функция.

Если A и B не получили одинаковый общий ключ, тогда почти наверняка эти ключи близки друг к другу, посредством удаления некоторого количества самых младших разрядов ключей, сгенерированные ключи почти всегда могут быть сделаны одинаковыми. A и B могут проверять, равны ли их общие ключи, посредством выполнения подтверждения ключа, например A может посылать B сообщение, содержащее пару (m, E(m)), где m - это сообщение, скажем, фиксированная строка или случайное число, а E(m) - шифровка, использующая общий ключ устройства A.

Посредством дешифровки E(m), используя общий ключ устройства B, B может проверить, равняются ли ключи. Если это так, B может отвечать A посредством информирования его о ситуации.

Если ключи не равны, A и B могут быть вовлечены в протокол уравнивания ключа. Например, они могут использовать тот факт, что два ключа арифметически близки друг к другу. Например, сетевые устройства A и B могут итеративно удалять самый младший разряд и отправлять сообщение подтверждения ключа до тех пор, пока ключи не будут равны. После достижения равных ключей A и B могут выполнять алгоритм получения ключа, чтобы вновь получить ключи с обычной длиной ключа.

Выбранные m частных модулей $p_1, p_2, \ldots, p_m$ предпочтительно попарно взаимно простые. Если эти числа попарно взаимно простые, нехватка совместимости между операциями по модулю увеличивается. Получение попарно взаимно простых чисел может быть осуществлено посредством выбора целых чисел по порядку, проверяя для каждого нового целого числа, являются ли все пары разных чисел все еще взаимно простыми, если нет, тогда выбранное число удаляется из набора. Эта процедура продолжается, пока не выбраны все m чисел.

Сложность увеличивается еще больше посредством требования, чтобы выбранные m частных модулей $p_1, p_2, \ldots, p_m$ являлись разными простыми числами. В этом случае можно требовать, чтобы каждое просто число имело форму $p_j=N+\gamma_j \cdot 2^b$. При этом γ_j - это целые числа, такие, что $|\gamma_j|<2^b$. Эксперименты подтвердили, что такие простые числа легкодоступны. Например, можно многократно выбирать случайное число γ_j и

проверять результирующее p_j, пока не будет найдено простое число. То же самое применяется, когда используется более общее выражение, как описано выше. Действительно, это следует из теоремы о простых числах для арифметических прогрессий, согласно которой, пока a имеет тот же порядок величины, что и b, в частности, для a<b, существует достаточно много таких простых чисел. В частности, для любой комбинации длины ключа в группе 64, 128, 196, 256 и степени в группе 2, 3 мы подтвердили посредством эксперимента, что много простых чисел этой формы могут быть сгенерированы, используя вышеупомянутый алгоритм, в течение практического временного интервала. При использовании простых чисел каждый полином f_j, таким образом, берется в конечном поле с p_j элементами.

Возможны многие варианты выбора разных параметров, используемых во время фазы регистрации и использования. Например, в упрощенном варианте осуществления частные модули меньше, чем публичные модули, и удовлетворяют соотношению $p_j=N-\beta_j\cdot 2^b$. При этом β_j - это положительные целые числа, такие, что $\beta_j<2^b$. Одним практичным способом выбора чисел, которые удовлетворяют этому требованию, является выбор набора из m случайных целых чисел β_j, таких, что $\beta_j<2^b$, расчет выбранных частных модулей из соотношения $p_j=N-\beta_j\cdot 2^b$. Как отмечалось, разность $Y_j-Z_j-\log_2(\zeta_j)$ может составлять a_jb. Подобным образом другие конструкции могут быть определены, следуя той же концепции. В частности, мы можем записать $p_j = \beta 2^X + \gamma_j 2^{Y_j} + \delta 2^W + \zeta_j 2^{Z_j}$ для частных модулей и $N = \beta 2^X + \delta 2^W$ для публичного модуля. Конкретной реализацией этой конструкции является $p_j = 2^{2(a+1)b} + \gamma_j 2^{(a+1)b} + 2^{ab} + \zeta_j$ и $N = 2^{2(a+1)b} + 2^{ab}$. В этом случае абсолютное значение членов γ_j и β_j·меньше чем 2^b и они ответственны за создание нелинейного эффекта на MSB и LSB коэффициентов локального сохраненного материала ключа на устройстве. Отметим, что, так как идентификаторы устройств имеют длину примерно b разрядов, $\gamma_j(\beta_j)$ влияет на MSB (LSB) коэффициентов долей полинома, оцениваемых в кольце целых чисел по модулю p_j. После этого, во время генерирования материала локального ключа для устройства, коэффициенты долей полинома в разных кольцах суммируются по целым числам, чтобы скрыть происхождение вкладов.

Ключ может быть сгенерирован следующим образом: $K_{AB} = < \frac{\sum_i < F_i^A(X)|_{X=B} >_{N_i}}{2^W} >_{2^b}$, но если используется еще более общее выражение для p_j и N, которое позволяет вводить нелинейные эффекты как MSB, так и в LSB, тогда деление после приведения по модулю N осуществляется на 2 в степени W, где 2^W - самая большая степень 2, являющаяся целым делителем N. Другие конструкции для N и p_j могут требовать деления на другую степень двойки. Так как двумерные полиномы в материале корневого ключа симметричны, общий ключ устройства A и общий ключ устройства B часто, хоть и не обязательно всегда, равны.

Фигура 1 - блок-схема, иллюстрирующая генератор 100 материала корневого ключа. Узел получения материала ключа сконфигурирован, чтобы предоставлять входные данные, кроме идентификационного номера, необходимые генератору материала локального ключа для генерирования материала локального ключа. Генератор ключа является примером узла получения материала ключа. Вместо генерирования всех

входных данных или их части некоторые параметры могут быть получены генератором материала корневого ключа посредством их приема; например узел получения ключа может содержать электронный приемник для приема входных данных, например публичного и частного модуля. Узел получения материала ключа получает все необходимые параметры, кроме идентификационного номера, из внешнего источника. В варианте осуществления a, b, m являются предопределенными, например принятыми, а публичный модуль и частные модули в наборах параметров и соответствующие (симметричные) двумерные полиномы генерируются. В варианте осуществления публичные модули тоже являются предопределенными, например принятыми.

Генератор 100 корневого ключа генерирует множество наборов параметров и содержит элемент 130 количества t наборов параметров, который содержит количество наборов параметров, которое должно быть сгенерировано. Например, t=2 или t=3, и т. д.

Генератор 100 корневого ключа содержит элемент 112 степени полинома, элемент 114 длины ключа и элемент 116 количества полиномов, сконфигурированные, чтобы предоставлять степень полинома, длину ключа и количество полиномов, например a, b и m соответственно, для заданного набора параметров. Обычно элемент 114 длины ключа будет одинаковым по всем наборам параметров. Обычно элемент 112 степени полинома также будет одинаковым по всем наборам параметров, хотя это не обязательно. В некоторых вариантах осуществления элемент 116 количества полиномов различается для разных наборов параметров; например, некоторые могут использовать m=1, в то время как некоторые могут использовать m=2. Наличие постоянного m, скажем, m=1 или m=2, для всех наборов также возможно.

Хотя эти элементы могут быть сгенерированы, например, в зависимости от обстоятельств, обычно эти параметры выбираются разработчиком системы. Например, элементы могут быть сконструированы в виде энергонезависимых элементов памяти или в виде приемников для приема значений элементов, или в виде энергозависимых элементов памяти, соединенных с приемником, и т. д. Подходящий выбор включает в себя t=2, a=2, b=128, m=2. Любое из этих чисел может быть увеличено или уменьшено, чтобы получить более или менее безопасную систему.

Генератор 100 корневого ключа содержит элемент 110 публичного модуля, сконфигурированный, чтобы предоставлять публичный модуль N набора параметров. Публичный модуль может выбираться или не выбираться разработчиком системы. Например, публичный модуль может являться набором удобных чисел, обеспечивающих быстрое приведение (близких или равных степени дойки). Публичный модуль выбирается в пределах диапазона, определенного посредством элементов 112 и 114.

Генератор 100 корневого ключа содержит узел 122 управления частным модулем, сконфигурированный, чтобы предоставлять частный модуль p, или несколько частных модулей $p_1, \ldots, p_m$. Например, они могут выбираться случайно в пределах подходящих границ.

Генератор 100 корневого ключа содержит узел 124 управления симметричным двумерным полиномом, сконфигурированный, чтобы предоставлять симметричный двумерный полином f, или несколько симметричных двумерных полиномов $f_1, \ldots, f_m$. Каждый симметричный двумерный полином выбирается с коэффициентами по случайному модулю в соответствии с частным модулем, то есть с частным модулем, имеющим тот же индекс. Коэффициенты могут выбираться в диапазоне от 0 до p-1, и они могут выбираться случайно.

Частные модули могут выбираться посредством прибавления к публичному модулю

или вычитания из него числа, кратного двойке в степени длины ключа. Это в результате даст такие частные модули, разница которых с публичным модулем заканчивается рядом последовательных нулей. Также можно выбирать публичный модуль и один или более частных модулей так, чтобы ряд последовательных нулей, имеющий длину ключа, появлялся не в конце, а в другой позиции, скажем, позиции ‘s’, считая с самого младшего разряда.

Фигура 1’ показывает пример материала 180 корневого ключа, сгенерированного посредством генератора 100 корневого ключа. Корневой материал 180 ключа содержит несколько наборов 140 параметров, в данном случае 3 набора. Корневой материал 180 ключа содержит три набора параметров. Первый набор параметров содержит публичный модуль 141, частные модули 151, 153 и 155 и соответствующие двумерные полиномы 152, 154 и 156. Второй набор параметров содержит публичный модуль 142, частные модули 161 и 163 и соответствующие двумерные полиномы 162 и 164. Третий набор параметров содержит публичный модуль 143, частные модули 171, 173 и 175 и соответствующие двумерные полиномы 172, 174 и 176. В данном случае степень полиномов скрыта в представлении полиномов, но она также может быть сделана явной. В данном примере материала 180 корневого ключа, длина 144 ключа также записывается.

Во время работы узел 100 получения материала корневого ключа многократно генерирует наборы параметров до тех пор, пока количество произведенных наборов не будет равно числу в элементе 130. Количество наборов параметров может быть записано в материале 180 корневого ключа в элементе 140.

Фигура 2 - блок-схема, иллюстрирующая генератор 200 материала локального ключа. Генератор 100 материала ключа и генератор 200 материала локального ключа формируют систему для конфигурирования сетевого устройства для использования общего ключа.

Генератор 200 материала локального ключа содержит устройство 240 манипулирования полиномом. Генератор 200 материала локального ключа содержит элемент 210 материала корневого ключа для предоставления материала корневого ключа устройству 240 манипулирования полиномом, то есть предоставления множества наборов параметров устройству 240 манипулирования полиномом, чтобы, в свою очередь, произвести множество одномерных полиномов. Элемент 210 может быть реализован посредством соответствующих элементов генератора 100 материала ключа; эти элементы также могут являться элементами памяти или шинами для соединения с генератором 100 материала ключа.

Генератор 200 материала локального ключа содержит генератор 260 запутывающих чисел для предоставления запутывающих чисел $'\epsilon_{A,i}'$ устройству 240 манипулирования полиномом. Запутывающее число может являться случайным числом, например, сгенерированным с помощью генератора случайных чисел. Генератор 260 запутывающих чисел может генерировать множество запутывающих чисел для множества коэффициентов одномерного полинома. Генератор 260 может быть ограничен генерированием одиночных чисел, например по одному запутывающему числу на набор параметров, или по одному запутывающему числу на по меньшей мере два набора параметров, но генератор 260 также может быть сконфигурирован, чтобы генерировать ненулевые запутывающие полиномы, которые должны добавляться к одномерному полиному, который соответствует текущему набору параметров, чтобы получить

запутанный одномерный полином. В варианте осуществления запутывающее число определяется для каждого коэффициента одномерного полинома. Запутывающий полином может иметь 1, 2 или более ненулевых коэффициентов.

Генератор 200 материала локального ключа содержит узел 250 управления сетевым устройством, сконфигурированный, чтобы принимать идентификационный номер, для которого должен быть сгенерирован материал локального ключа, например, от сетевого устройства, и сконфигурирован, чтобы передавать материал локального ключа на сетевое устройство, соответствующее идентификационному номеру. Вместо приема идентификационного номера, он также может быть сгенерирован, например, в виде случайного, серийного или одноразового номера. В последнем случае идентификационный номер передается вместе с материалом локального ключа на сетевое устройство.

Устройство 240 манипулирования полиномом генерирует одномерный полином для каждого набора параметров в элементе 210 материала корневого ключа.

Для каждого набора параметров устройство 240 манипулирования полиномом получает, возможно, несколько, одномерных полиномов посредством подстановки идентификационного номера от узла 250 управления в каждый из двумерных полиномов и приведения каждого по модулю соответствующего частного модуля. Несколько результирующих приведенных одномерных полиномов суммируются по коэффициентам с помощью натурального арифметического сложения. Также прибавляется одно или более запутывающих чисел. Предпочтительно результат приводится, опять же по коэффициентам, по модулю публичного модуля, коэффициенты последнего могут быть представлены в диапазоне от 0 до N-1.

Запутывающие одномерные полиномы являются частью материала локального ключа, соответствующего идентификационному номеру. При необходимости публичный модуль, степень и длина ключа также передаются на сетевое устройство.

Фигура 2' показывает локальный материал 280 корневого ключа, сгенерированный для сетевого устройства из материала 180 корневого ключа. Локальный материал 280 корневого ключа содержит несколько наборов 140 параметров (здесь, 3), длину ключа 144, публичные модули 141, 142 и 143 и соответствующие сгенерированные (возможно, запутанные) одномерные полиномы 252, 262 и 274, соответственно. Необязательно, локальный материал 280 корневого ключа может содержать степень 2 для деления и ключевой модуль для генерирования общего ключа.

Фигура 3 - блок-схема, иллюстрирующая сеть 300 связи, содержащую несколько сетевых устройств; показаны первое сетевое устройство 310 и второе сетевое устройство 320. Мы проиллюстрируем первое сетевое устройство 310. Второе сетевое устройство 320 может быть таким же, или работать по тем же принципам.

Сетевое устройство 310 содержит приемопередатчик 330, объединяющий передатчик и приемник для передачи на второе сетевое устройство 320 и приема от него сообщений в электронном, например цифровом, формате, проводным или беспроводным образом. Возможно, приемопередатчик 330 также используется, чтобы принимать материал локального ключа от сетевой службы 200. С помощью приемопередатчика 330 принимается идентификационный номер другого устройства; на фигуре, второго устройства 320.

Сетевое устройство 310 содержит узел 344 получения материала локального ключа. Узел 344 получения материала локального ключа может быть реализован в виде локальной памяти, например энергонезависимой памяти, такой как флэш-память, для хранения материала локального ключа. Узел 344 получения материала локального

ключа также может быть сконфигурирован, чтобы получать материал локального ключа от генератора 200, например, через приемопередатчик 330. Узел 344 получения материала локального ключа сконфигурирован, чтобы обеспечивать устройство манипулирования полиномом необходимыми параметрами.

Сетевое устройство 310 содержит устройство 342 манипулирования полиномом. Устройство 342 манипулирования полиномом работает в две фазы.

На фазе подстановки идентификационный номер второго сетевого устройства подставляется (530) в каждый из одномерных полиномов в материале локального ключа. Результат подстановки приводится по модулю публичного модуля, соответствующего упомянутому одномерному полиному. На последующей фазе сложения, результаты приведений по модулю публичного модуля суммируются и приводятся (540) по модулю ключевого модуля. Отметим, что для некоторых комбинаций N и частного модуля, деление на степень 2 требуется перед тем, как результат будет приведен по модулю ключевого модуля.

Сетевое устройство 310 содержит устройство 346 получения ключа для получения общего ключа из результата приведения по модулю ключевого модуля. Например, устройство 346 получения ключа может удалять один или более самых младших разрядов. Устройство 346 получения ключа может также использовать функцию получения ключа. Также возможно использовать результат второго приведения без дальнейшей обработки.

Сетевое устройство 310 содержит опциональный узел 348 уравнивания ключа. Отметим, что может случиться так, что общий ключ, полученный в первом сетевом устройстве, не равняется ключу, полученному во втором устройстве (на основании идентификационного номера первого сетевого устройства). Если это считается нежелательным, можно последовать протоколу уравнивания.

Сетевое устройство 310 содержит криптографический элемент 350, сконфигурированный, чтобы использовать общий ключ для криптографического применения. Например, криптографический элемент 350 может зашифровывать или аутентифицировать подлинность сообщения первого сетевого устройства с помощью общего ключа перед передачей его на второе сетевое устройство, например сообщения о статусе. Например, криптографический элемент 350 может расшифровывать или проверять подлинность сообщения, принятого от второго сетевого устройства.Обычно система 200 для конфигурирования сетевого устройства для использования общего ключа, и первое сетевое устройство 310, сконфигурированное, чтобы определять общий ключ, содержат микропроцессор (не показан), который выполняет подходящее программное обеспечение, хранящееся на соответствующих устройствах, например, это программное обеспечение может быть загружено и сохранено в соответствующей памяти, например ОЗУ (RAM) (не показано).

Интересный вариант осуществления получается для a=1, особенно в комбинации с более высокими значениями m, скажем, более высокими чем 1, 2 или больше, 4 или больше. Необходимое манипулирование полиномом сводится к одному умножению и приведению, обеспечивая особенно простую реализацию. Однако даже для такого простого случая восстановление изначальных двумерных полиномов не является простой задачей, и становится еще более сложным с более высокими значениями m. Несмотря на то что нет известных жизнеспособных атак даже при a=1, линейная конструкция может являться стартовой точкой для будущего анализа, по этой причине можно пожелать ввести ограничение $a > 1$.

Фигура 4 - блок-схема последовательности операций, иллюстрирующая способ 400

генерирования материала локального ключа. Способ 400 может быть использован третьей доверенной стороной. На этапе 410 получаются требуемые параметры. В частности, получаются несколько наборов параметров, по меньшей мере два. Каждый набор параметров содержит публичный модуль и по меньшей мере один частный модуль, и по меньшей мере один двумерный полином. На этапе 420 получается идентификационный номер сетевого устройства, например, по сети связи. Идентификационный номер может быть принят в электронном сообщении.

Этап 430 повторяется по одному разу для каждого набора параметров. Полученный идентификационный номер подставляется в двумерный полином и приводится по модулю частного модуля. Может быть больше, например, 2 двумерных полинома. В этом случае подстановка производится в каждом из них, и результаты складываются с помощью целочисленной арифметики. На этапе 440 результат запутывается, например, посредством добавления запутывающего полинома. В простой реализации запутывание может являться одиночным коэффициентом. Этап 440 является необязательным. Таким образом, или как описано в материалах настоящей заявки, получается комбинация одномерного полинома и публичного модуля, которая может формировать часть материала локального ключа. На этапе 450 решается, остались ли еще наборы параметров, и, если это так, этапы 430 и 440 повторяются для следующего набора параметров. На этапе 450 материал локального ключа, включающий в себя запутанные одномерные полиномы, сохраняется на сетевом устройстве.

Фигура 5 - блок-схема последовательности операций, иллюстрирующая способ 500 генерирования общего ключа. Способ 500 может выполняться сетевым устройством.

На этапе 510 получается внешний идентификационный номер другого сетевого устройства, например, посредством приема электронного сообщения. На этапе 520 локальный идентификационный номер передается на другое сетевое устройство. После этапов 510 и 520, локальное сетевое устройство и внешнее сетевое устройство имеют идентификационный номер друг друга. Используя материал локального ключа, они продолжают получать общий ключ.

Локальное сетевое устройство повторяет этап 530 подстановки для одномерного полинома в своем материале локального ключа. На этапе 530 внешний идентификационный номер подставляется в запутанный одномерный полином по модулю соответствующего публичного модуля. На этапе 535 решается, остались ли еще одномерные полиномы, и, если это так, этап 530 повторяется для одномерного полинома материала локального ключа. На этапе 540 результаты приведений по модулю публичного модуля суммируются и приводятся по модулю ключевого модуля.

Результат этапа 550 является началом получения общего ключа. На этапе 550 получают общий ключ, например, посредством применения алгоритма получения ключа. На этапе 560 сообщение подтверждения ключа передается на другое сетевое устройство, и, на этапе 570 определяется, подтвержден ли ключ. Если ключ не подтвержден на этапе 570, тогда способ продолжается на этапе 550 с получением нового ключа. Например, этап 550 может удалять один дополнительный самый младший разряд каждый раз, когда ключ не подтверждается. Если ключ подтвержден, он может использоваться в произвольном криптографическом применении или сохраняться локально для последующего использования.

Этапы 550, 560 и 570 вместе формируют протокол уравнивания ключа. Например, на этапе 560 одноразовое число и шифровка одноразового числа с использованием общего ключа, полученного на этапе 550, могут передаваться на второе устройство. На этапе 560 принимается сообщение от второго устройства. Принятое сообщение

может просто говорить, что принятое сообщение подтверждения ключа показало, что ключи не равны. Принятое сообщение может также содержать сообщение подтверждения ключа. В последнем случае первое сетевое устройство проверяет сообщение подтверждения ключа и устанавливает, равны ли ключи. Если не равны, получают новый ключ, например, посредством удаления самого младшего разряда.

Фигура 6 показывает схематическую форму возможной последовательности сообщений между двумя сетевыми устройствами, устройствами A и B, в то время как они генерируют общий ключ. Время идет вниз. На этапе 610 сетевое устройство A передает свой идентификационный номер на устройство B. На этапе 620 устройство B передает свой идентификационный номер и сообщение подтверждения ключа для общего ключа (K1), который оно получило на основании идентификационного номера A и своего материала локального ключа. На этапе 630 устройство A обнаружило, что они не сгенерировали одинаковый ключ. Устройство A удаляет один самый младший разряд (скажем, посредством целочисленного деления на 2), чтобы получить ключ K2. На этапе 630 устройство A передает новое сообщение подтверждения ключа. Таким образом, A и B обмениваются сообщениями 640 подтверждения ключа до тех пор, пока они не придут к одинаковому ключу на этапе 650. На этапе 650 устройство A передает сообщение подтверждения ключа на устройство B. Устройство B смогло подтвердить, что они пришли к одинаковому ключу. На этапе 660 оно передает подтверждение этого факта, это может быть аутентифицированное сообщение или сообщение подтверждения ключа, и т.д. На этапе 670 устройство A передает сообщение M1, которое зашифровано (скажем, используя AES (Advanced Encryption Standard, улучшенный стандарт шифрования)) и/или аутентифицировано (скажем, используя HMAC (Hashed Message Authentication Code, хэш-код аутентификации сообщений)), используя новый равный общий ключ.

Будет приниматься во внимание, что изобретение также распространяется на компьютерные программы, в частности, компьютерные программы на или в носителе, приспособленные для осуществления изобретения на практике. Программа может быть представлена в виде исходного кода, объектного кода, промежуточного источника кода и объектного кода, такого как частично скомпилированная форма, или в любом другом виде, пригодном для использования при реализации способа согласно изобретению. Вариант осуществления, относящийся к компьютерному программному продукту, содержит исполняемые компьютером команды, соответствующие каждому из этапов обработки по меньшей мере одного из изложенных способов. Эти программы могут быть подразделены на подпрограммы и/или сохранены в одном или более файлах, которые могут быть связаны статически или динамически. Другой вариант осуществления, относящийся к компьютерному программному продукту, содержит исполняемые компьютером команды, соответствующие каждому из средств по меньшей мере одной из изложенных систем и/или продуктов.

Стоит отметить, что вышеупомянутые варианты осуществления скорее иллюстрируют, чем ограничивают изобретение, и что специалисты в данной области техники будут способны сконструировать многие альтернативные варианты осуществления.

В формуле изобретения любые символы ссылок, помещенные между круглыми скобками, не должны истолковываться в качестве ограничивающих формулу изобретения. Использование глагола «содержать» и его спряжений не исключает наличия элементов или этапов, отличных от изложенных в пункте формулы изобретения. Использование единственного числа при описании элемента не исключает наличия

множества таких элементов. Изобретение может быть реализовано посредством аппаратных средств, содержащих несколько отдельных элементов, и посредством подходящим образом запрограммированного компьютера. В пункте формулы изобретения типа устройства, перечисляющем несколько средств, некоторые из этих средств могут быть воплощены посредством одного и того же элемента аппаратного обеспечения. Простое обстоятельство, что определенные меры перечислены во взаимно разных зависимых пунктах формулы изобретения, не служит признаком того, что комбинация этих мер не может быть использована с выгодой.

Список номеров ссылок для фигур 1-3:	
100	узел получения материала корневого ключа
110	узел управления публичным модулем
112	элемент степени полинома
114	элемент длины ключа
116	элемент количества полиномов
122	узел управления частным модулем
124	узел управления симметричным двумерным полиномом
130, 140	двумерный полином
180	материал корневого ключа
200	генератор материала локального ключа
210	элемент материала корневого ключа
240	устройство манипулирования полиномом
250	узел управления сетевым устройством
252, 262, 272	одномерный полином
260	генератор запутывающих чисел
300	сеть связи
310	первое сетевое устройство
320	второе сетевое устройство
330	приемопередатчик
342	устройство манипулирования полиномом
344	узел получения материала локального ключа
346	устройство получения ключа
348	узел уравнивания ключа
350	криптографический элемент

(57) Формула изобретения

1. Способ конфигурирования сетевого устройства для использования общего ключа, содержащий этапы, на которых:

- получают в электронной форме по меньшей мере два набора параметров, каждый набор параметров содержит частный модуль, публичный модуль и двумерный полином, имеющий целые коэффициенты, бинарное представление публичного модуля и бинарное представление частного модуля являются одинаковыми в имеющих по меньшей мере длину ключа последовательных разрядах,

- генерируют материал локального ключа для сетевого устройства, что содержит этапы, на которых

- получают в электронной форме идентификационный номер для сетевого устройства, и

- для каждого набора параметров получают соответствующий одномерный полином посредством определения, используя устройство манипулирования полиномом, одномерного полинома из двумерного полинома набора параметров посредством подстановки идентификационного номера в упомянутый двумерный полином и приведения результата подстановки по модулю частного модуля набора параметров,

и

- электронным образом сохраняют на сетевом устройстве сгенерированный материал локального ключа, содержащий публичный модуль упомянутого каждого набора параметров и соответствующий одномерный полином упомянутого каждого набора параметров.

2. Способ по п. 1, в котором генерирование материала локального ключа для сетевого устройства содержит этапы, на которых

для по меньшей мере двух из по меньшей мере двух наборов параметров

- генерируют ненулевой запутывающий полином, соответствующий упомянутому набору параметров,

- прибавляют (440), используя устройство для манипулирования полиномом, ненулевой запутывающий полином к одномерному полиному, который соответствует упомянутому набору параметров, чтобы получить запутанный одномерный полином,

- при этом сгенерированный материал локального ключа содержит запутанный одномерный полином.

3. Способ по п. 2, в котором каждый коэффициент суммы запутывающих полиномов кратен 2 в степени длины ключа.

4. Способ по п. 2, в котором каждый коэффициент суммы запутывающих полиномов, разделенный на степень двойки, округленный вниз до целого числа, кратен 2 в степени длины ключа.

5. Способ по п. 1, в котором все двумерные полиномы во всех наборах параметров являются симметричными полиномами.

6. Способ по п. 1, в котором во всех наборах параметров одинаковые имеющие по меньшей мере длину ключа последовательные разряды бинарного представления публичного модуля соответствующего набора параметров такие же, как и самые младшие имеющие длину ключа разряды частного модуля соответствующего набора параметров.

7. Способ по п. 6, в котором имеющие по меньшей мере длину ключа последовательные разряды являются самыми младшими имеющими длину ключа разрядами.

8. Способ по п. 1, содержащий этапы, на которых:

- генерируют частный модуль, используя электронный генератор случайных чисел, или

- генерируют двумерный полином, используя электронный генератор случайных чисел, посредством генерирования одного или более случайных коэффициентов для двумерного полинома.

9. Способ по п. 1, в котором один или все публичные модули удовлетворяют $2^{(a+2)b-1} \leq N$, где N представляет публичный модуль, а представляет степень двумерного полинома, а b представляет длину ключа.

10. Способ по п. 1, в котором по меньшей мере два набора параметров содержат множество частных модулей и множество двумерных полиномов, имеющих коэффициенты по модулю, таких, что имеется набор имеющих длину ключа последовательных позиций, в которых бинарное представление публичного модуля согласуется с бинарным представлением всех частных модулей, и способ

дополнительно содержит

определение одномерного полинома, которое содержит подстановку идентификационного номера в каждый из множества двумерных полиномов, приведение по модулю частного модуля из множества частных модулей, соответствующих одному

симметричному двумерному полиному, и сложение множества результатов множества приведений.

11. Способ по п. 1, в котором запутывающее число генерируется так, что $|\epsilon^{A}_{i,k}| < 2^{(a+2-k)b-2}$, где $\epsilon_{A,i}$ обозначает запутывающее число, i обозначает степень одночлена, соответствующего коэффициенту, а представляет степень двумерного полинома и b представляет длину ключа.

12. Способ определения общего ключа для первого сетевого устройства, причем ключ является криптографическим ключом, и способ содержит этапы, на которых

- получают материал локального ключа для первого сетевого устройства в электронной форме, материал локального ключа содержит по меньшей мере два одномерных полинома и соответствующие публичные модули,

- получают идентификационный номер для второго сетевого устройства, причем второе сетевое устройство отлично от первого сетевого устройства,

- для каждого из по меньшей мере двух одномерных полиномов подставляют (530) идентификационный номер второго сетевого устройства в упомянутый одномерный полином и приводят результат подстановки по модулю публичного модуля, соответствующего упомянутому одномерному полиному, и

- суммируют результаты приведений по модулю публичного модуля и приводят по модулю ключевого модуля, и

- получают общий ключ из результата приведения по модулю ключевого модуля.

13. Способ по п. 12, содержащий этапы, на которых

- определяют, получили ли первое сетевое устройство и второе сетевое устройство одинаковый общий ключ или нет, и если нет, получают дополнительный общий ключ из результата приведения по модулю ключевого модуля.

14. Способ по п. 12, содержащий деление результата подстановки по модулю публичного модуля на делитель строки нулевого разряда, который является степенью двойки, при этом делитель строки нулевого разряда больше чем 1, и округление вниз результата деления до целого числа.

15. Сетевое устройство, сконфигурированное, чтобы определять общий ключ, являющийся криптографическим ключом, причем сетевое устройство содержит

- узел получения материала локального ключа, сконфигурированный, чтобы получать материал локального ключа для сетевого устройства в электронной форме, материал локального ключа содержит по меньшей мере два одномерных полинома и соответствующие публичные модули,

- приемник, сконфигурированный, чтобы получать идентификационный номер для другого дополнительного сетевого устройства,

- устройство манипулирования полиномом, сконфигурированное, чтобы для каждого из по меньшей мере двух одномерных полиномов подставлять идентификационный номер второго сетевого устройства в упомянутый одномерный полином, приводить результат подстановки по модулю публичного модуля, соответствующего упомянутому одномерному полиному, и суммировать результаты приведений по модулю публичного модуля и приводить по модулю ключевого модуля, и

- устройство получения ключа, сконфигурированное, чтобы получать общий ключ из результата приведения по модулю ключевого модуля.

16. Система для конфигурирования сетевого устройства для использования общего ключа, содержащая

- узел получения материала ключа для получения в электронной форме по меньшей мере двух наборов параметров, набор параметров, содержащий частный модуль,

публичный модуль и двумерный полином, имеющий целые коэффициенты, бинарное представление публичного модуля и бинарное представление частного модуля являются одинаковыми в имеющих по меньшей мере длину ключа последовательных разрядах,

- генератор для генерирования материала локального ключа для сетевого устройства, содержащий

- узел управления сетевым устройством для получения в электронной форме идентификационного номера для сетевого устройства и для электронного сохранения сгенерированного материала локального ключа на сетевом устройстве,

- устройство манипулирования полиномом, сконфигурированное для получения для каждого набора параметров соответствующего одномерного полинома посредством определения одномерного полинома из двумерного полинома набора параметров посредством подстановки идентификационного номера в упомянутый двумерный полином и приведения результата подстановки по модулю частного модуля набора параметров.

17. Невременный машиночитаемый носитель данных, содержащий программу, содержащую команды для предписания процессору выполнять способ конфигурирования сетевого устройства для использования общего ключа, содержащий этапы, на которых:

получают в электронной форме по меньшей мере два набора параметров, каждый набор параметров содержит частный модуль, публичный модуль и двумерный полином, имеющий целые коэффициенты, бинарное представление публичного модуля и бинарное представление частного модуля являются одинаковыми в имеющих по меньшей мере длину ключа последовательных разрядах,

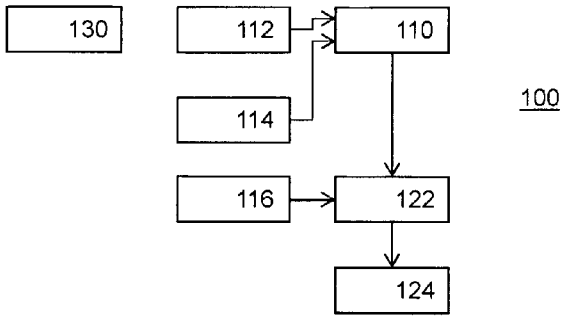
- генерируют материал локального ключа для сетевого устройства, что содержит этапы, на которых

получают в электронной форме идентификационный номер для сетевого устройства, и

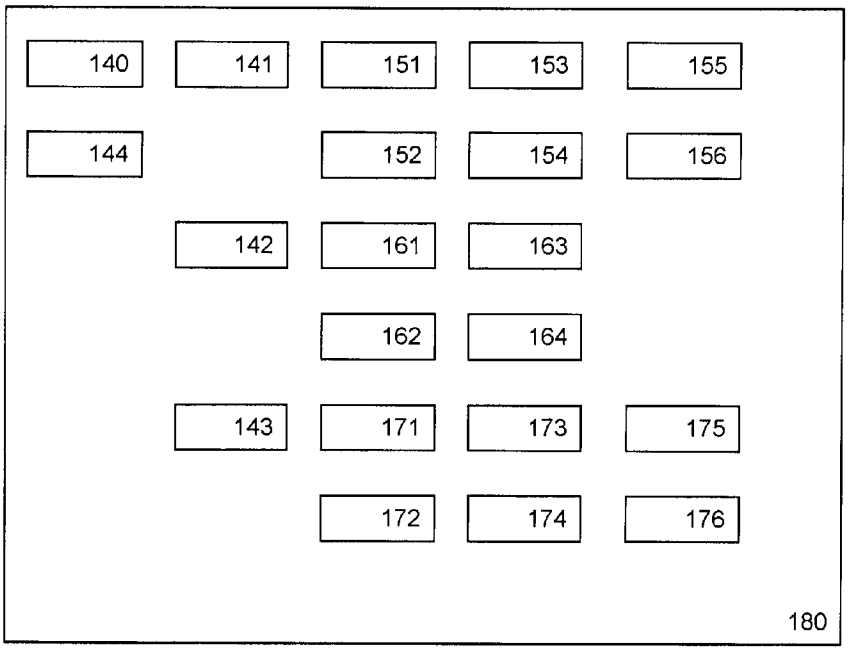
для каждого набора параметров получают соответствующий одномерный полином посредством определения, используя устройство манипулирования полиномом, одномерного полинома из двумерного полинома набора параметров посредством подстановки идентификационного номера в упомянутый двумерный полином и приведения результата подстановки по модулю частного модуля набора параметров, и

электронным образом сохраняют на сетевом устройстве сгенерированный материал локального ключа, содержащий публичный модуль упомянутого каждого набора параметров и соответствующий одномерный полином упомянутого каждого набора параметров.

1

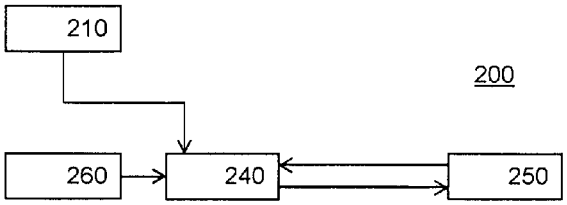

130
112
110
114
100
116
122
124

ФИГ.1

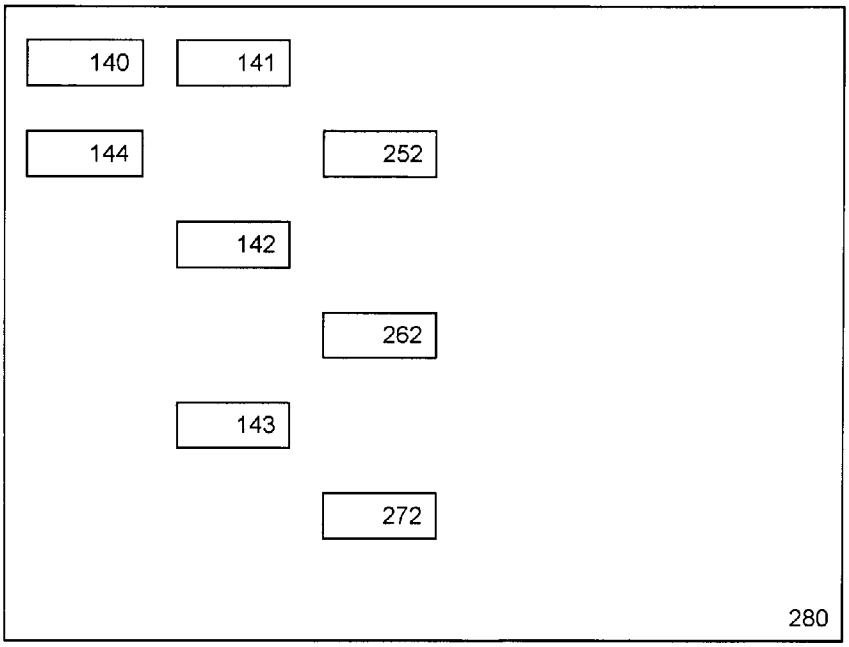

140
141
151
153
155
144
152
154
156
142
161
163
162
164
143
171
173
175
172
174
176
180

ФИГ.1'

2

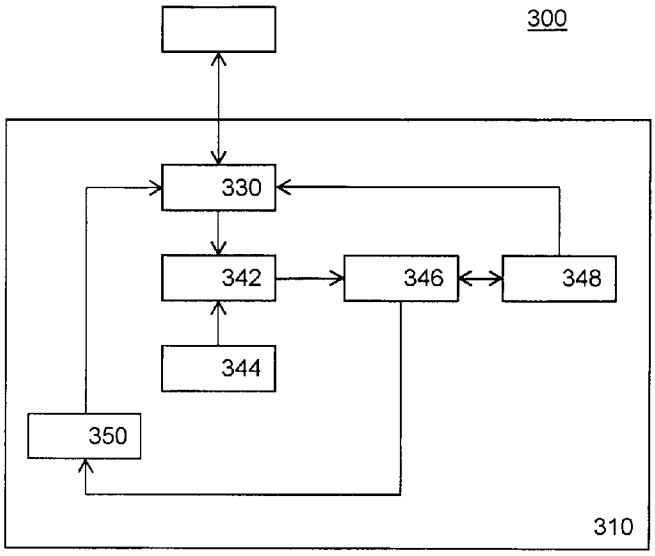


ФИГ.2



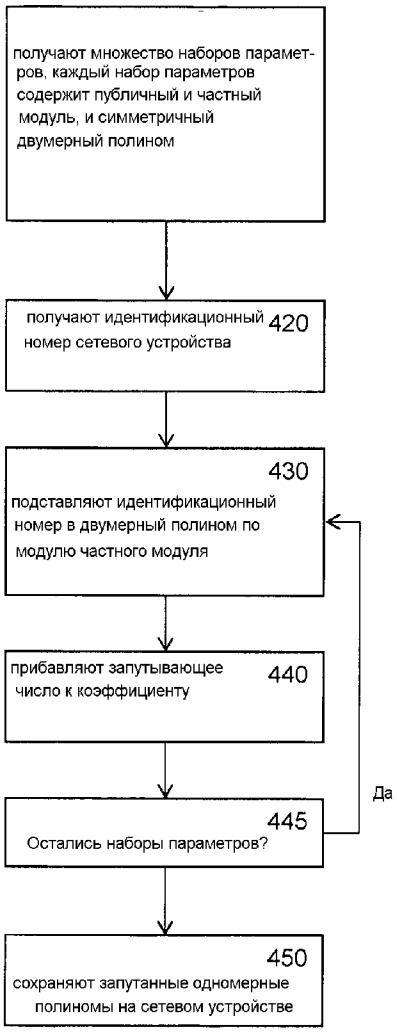
ФИГ.2'

3

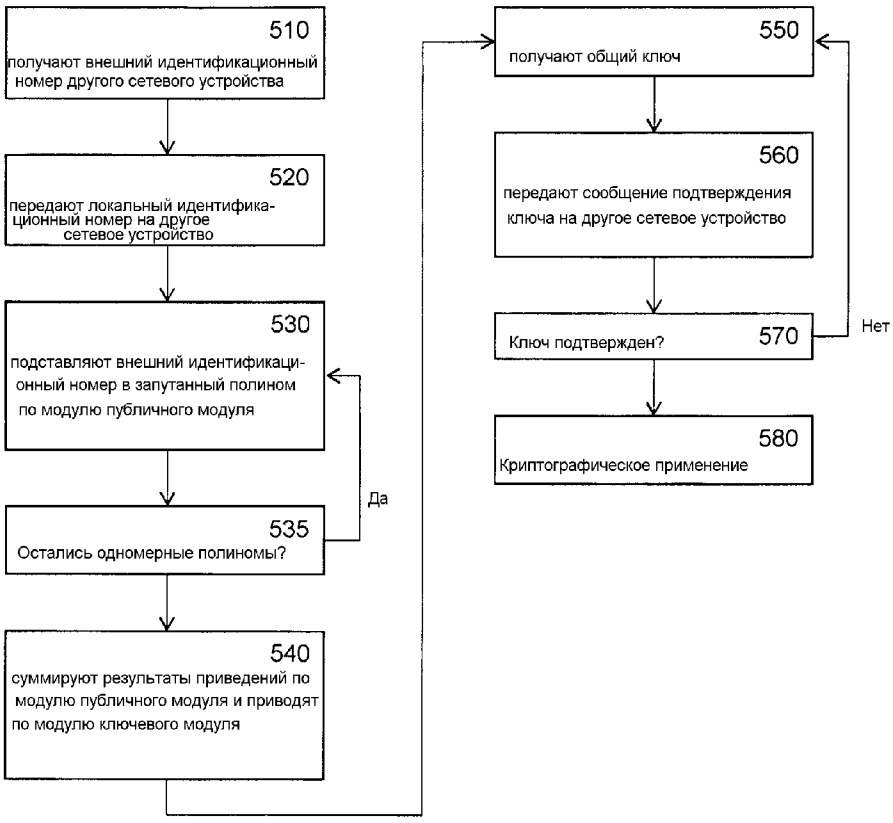


ФИГ.3

4

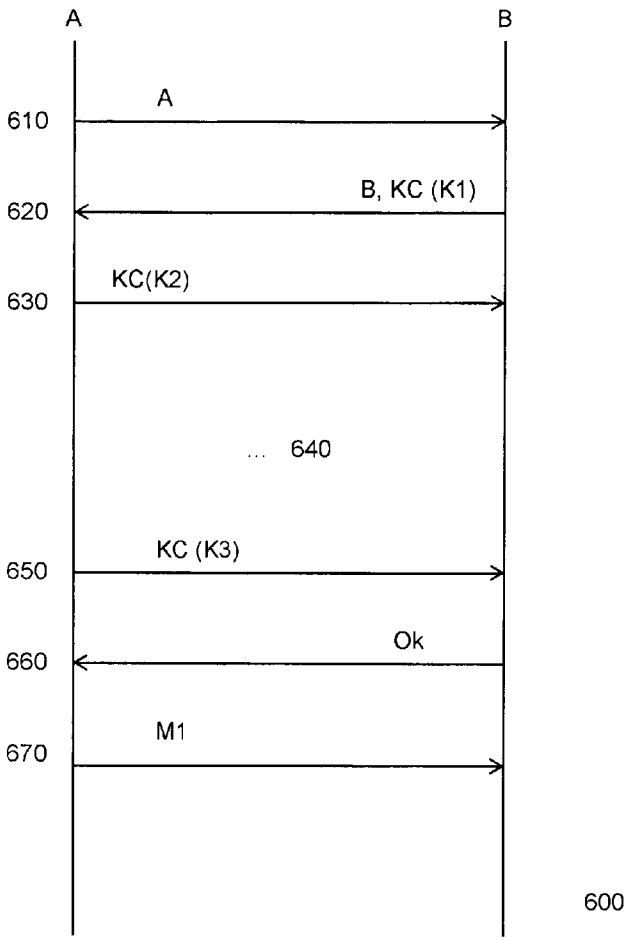

400
получают множество наборов параметров, каждый набор параметров содержит публичный и частный модуль, и симметричный двумерный полином
получают идентификационный номер сетевого устройства
420
430
подставляют идентификационный номер в двумерный полином по модулю частного модуля
прибавляют запутывающее число к коэффициенту
440
Да
445
Остались наборы параметров?
450
сохраняют запутанные одномерные полиномы на сетевом устройстве

ФИГ.4


510
получают внешний идентификационный номер другого сетевого устройства
520
передают локальный идентификационный номер на другое сетевое устройство
530
подставляют внешний идентификационный номер в запутанный полином по модулю публичного модуля
535
Остались одномерные полиномы?
Да
540
суммируют результаты приведений по модулю публичного модуля и приводят по модулю ключевого модуля
550
получают общий ключ
560
передают сообщение подтверждения ключа на другое сетевое устройство
570
Ключ подтвержден?
Нет
580
Криптографическое применение

ФИГ.5

6



ФИГ.6