РОССИЙСКАЯ ФЕДЕРАЦИЯ

(19) **RU** (11) **2 357 365** (13) **C2**

(51) МПК
H04L 9/06 (2006.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ,
ПАТЕНТАМ И ТОВАРНЫМ ЗНАКАМ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21), (22) Заявка: **2007101704/09, 06.06.2005**

(24) Дата начала отсчета срока действия патента:
06.06.2005

(30) Конвенционный приоритет:
18.06.2004 FR 0406678

(43) Дата публикации заявки: **27.07.2008**

(45) Опубликовано: **27.05.2009** Бюл. № **15**

(56) Список документов, цитированных в отчете о поиске: **WO 03101020 A1, 04.12.2003. RU 2129336 C1, 20.04.1999. GB 2345229, 28.06.2000. US 2002027987 A1, 07.03.2002.**

(85) Дата перевода заявки PCT на национальную фазу: **18.01.2007**

(86) Заявка PCT:
FR 2005/001376 (06.06.2005)

(87) Публикация PCT:
WO 2006/008355 (26.01.2006)

Адрес для переписки:
103735, Москва, ул.Ильинка, 5/2, ООО "Союзпатент", пат.пов. С.В.Истомину

(72) Автор(ы):
ДОТТА Эмманюэлль (FR),
ШАБАНН Эрве (FR),
КАРЛЬЕ Венсан (FR)

(73) Патентообладатель(и):
САЖЕМ СЕКЮРИТЕ (FR)

RU 2357365 C2

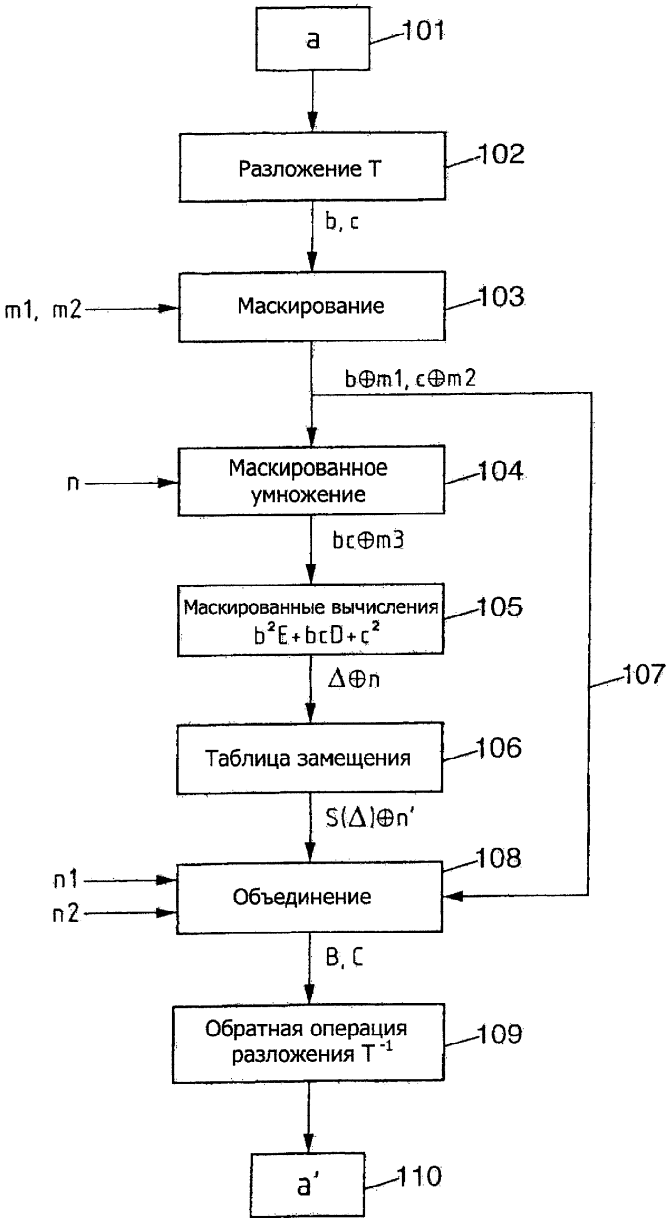
(54) СПОСОБ И УСТРОЙСТВО ДЛЯ ВЫПОЛНЕНИЯ КРИПТОГРАФИЧЕСКОГО ВЫЧИСЛЕНИЯ

(57) Реферат:

Изобретение относится к области криптографии. Технический результат заключается в уменьшении требуемой памяти для выполнения шифрования. Сущность изобретения заключается в том, что криптографическое вычисление осуществляют в электронном компоненте согласно определенному криптографическому алгоритму, включающему, по меньшей мере, одну точно определенную нелинейную операцию на k-битовых блоках данных, при этом k является целым числом больше 2. Генерируют несколько промежуточных маскированных блоков данных из j бит (b⊕m, c⊕m2, Δ⊕n) на основании исходного блока данных (a) из k бит, при этом j является целым числом, меньшим k. Затем производят нелинейную операцию S, по меньшей мере, на одном j-битовом маскированном промежуточном блоке данных (Δ⊕n) при помощи таблицы замещения (106) с 2^j входами, получая j-битовый измененный блок данных (S(Δ)⊕n). Измененный j-битовый блок данных объединяют, по меньшей мере, с некоторыми из указанных j-битовых маскированных промежуточных блоков данных в один итоговый k-битовый блок (a'),

RU 2357365 C2

соответствующий исходному k-битовому блоку данных, через преобразование, включающее указанную точно определенную нелинейную операцию. 2 н. и 17 з.п. ф-лы, 2 ил.



Фиг. 1

RU 2357365 C2

RU 2357365 C2

RUSSIAN FEDERATION

FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY,
PATENTS AND TRADEMARKS

(19) **RU** (11) **2 357 365** (13) **C2**

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

(12) ABSTRACT OF INVENTION

(21), (22) Application: **2007101704/09, 06.06.2005**

(24) Effective date for property rights:
06.06.2005

(30) Priority:
18.06.2004 FR 0406678

(43) Application published: **27.07.2008**

(45) Date of publication: **27.05.2009 Bull. 15**

(85) Commencement of national phase: **18.01.2007**

(86) PCT application:
FR 2005/001376 (06.06.2005)

(87) PCT publication:
WO 2006/008355 (26.01.2006)

Mail address:
103735, Moskva, ul.Il'inka, 5/2, OOO "Sojuzpatent", pat.pov. S.V.Istominu

(72) Inventor(s):
DOTTA Ehmmanjuehll' (FR),
ShABANN Ehrve (FR),
KARL'E Vensan (FR)

(73) Proprietor(s):
SAZhEM SEKJuRITE (FR)

RU 2357365 C2

(54) METHOD AND DEVICE FOR CARRYING OUT CRYPTOGRAPHIC COMPUTATION

(57) Abstract:

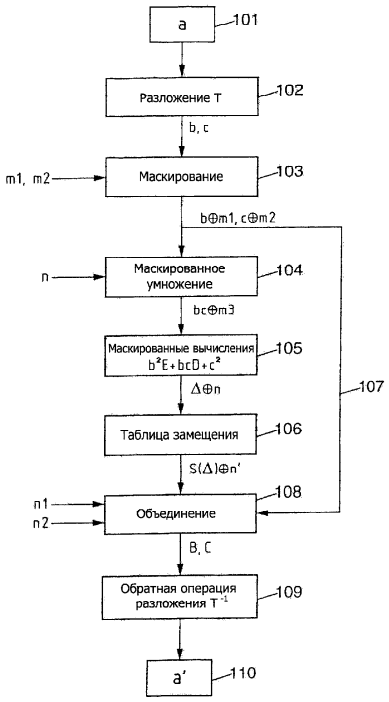
FIELD: information technology.

SUBSTANCE: present invention relates to cryptography. The essence of the invention lies in the fact that cryptographic computation is carried out in an electronic component according to a particular cryptographic algorithm, which includes at least one specific nonlinear operation on k-bit data blocks, where k is an integer greater than 2. Several masked intermediate data blocks from j bits (b⊕m, c⊕m2, Δ⊕n) based on the initial data block (a) from k bits are generated, where j is an integer less than k. Then a nonlinear operation S is carried out on at least one j-bit masked intermediate data block (Δ⊕n) with the help of a substitution table (106) with 2^j two inputs, to obtain a j-bit changed data block (S(Δ)⊕n). The changed j-bit data block is joined with at least several of the indicated j-bit masked intermediate data blocks into one final k-bit block (a'), corresponding to the initial k-bit data block, through transformation, which includes the indicated specific nonlinear operation.

EFFECT: reduction in the memory required for encryption.

19 cl, 2 dwg

RU 2357365 C2



Фиг. 1

RU 2357365 C2

RU 2357365 C2

Настоящее изобретение относится к области криптографии и, в частности, к защите секретности ключей, используемых для криптографических алгоритмов. В дальнейшем его описание связано с неограничительным применением для выполнения шифрования или дешифрования.

Алгоритмы шифрования, соответственно дешифрования, называемые еще алгоритмами криптографической защиты, предназначены для шифрования и соответственно дешифрования информации. Как правило, такие алгоритмы содержат последовательность из нескольких операций или вычислений, которые последовательно производят над предназначенными для шифрования данными для получения зашифрованных данных.

Такие алгоритмы шифрования могут подвергаться «атаке» с целью нарушения секретности используемых ключей. В настоящее время известны многие типы атак.

Так, некоторые атаки основаны на утечке информации, обнаруживаемой во время выполнения алгоритмов шифрования. В основном эти атаки основаны на корреляции между утечками информации, обнаруживаемыми во время обработки данных и ключа или ключей алгоритмом шифрования (атаки путем анализа потребления тока, электромагнитных излучений, времени вычисления и т.д.).

Известны методы защиты от таких атак. Одним из таких наиболее используемых методов защиты является случайное маскирование промежуточных данных, обрабатываемых алгоритмом шифрования или дешифрования. В таком типе защиты данные на входе маскируются случайными значениями. Таким образом, промежуточные данные, получаемые в результате операций, осуществляемых в алгоритме, могут быть декоррелированы от ключа или ключей.

Атаки с целью нарушения секретности ключей алгоритма шифрования аналогичны атакам с целью нарушения секретности ключей алгоритма дешифрования. В дальнейшем признаки, описанные для алгоритма шифрования, относятся также и к алгоритму дешифрования.

Как правило, алгоритм шифрования содержит несколько линейных и/или нелинейных операций. Для шифрования исходных данных получают промежуточные данные после каждой из операций алгоритма шифрования. При манипулировании маскированными промежуточными данными после каждой операции получают маскированные промежуточные данные. Таким образом получают защиту алгоритма шифрования.

Вместе с тем после каждой из этих операций имеет смысл извлечь немаскированные промежуточные данные путем «демаскирования» данных. Легче демаскировать промежуточные данные, полученные в результате линейной операции. Действительно, линейную операцию L, производимую на данных x, маскированных при помощи единицы или исключения со случайной маской m, можно записать следующим образом:

$$L(x \oplus m)=L(x) \oplus L(m).$$

Таким образом, зная m, можно легко демаскировать $L(x \oplus m)$, чтобы получить L(x).

Совсем иначе обстоит дело с нелинейными операциями. Действительно, для нелинейной операции F, производимой на данных x, маскированной при помощи единицы или исключения со случайной маской m, можно записать:

$$F(x \oplus m)=F(x) \oplus F(m).$$

Чтобы демаскировать промежуточные данные, обработанные алгоритмом шифрования, необходимо осуществить ряд вычислений, которые могут быть сложными и дорогостоящими в зависимости от защищаемого алгоритма шифрования.

Известны алгоритмы шифрования, в которых используются нелинейные операции, такие как алгоритмы DES от "Data Encryption Standard" или алгоритм AES от "Advanced Encryption Standard". Было предложено несколько способов защиты алгоритма AES при помощи маскирования.

В таких алгоритмах нелинейные операции, как правило, осуществляют в виде таблиц замещения. Так, нелинейную операцию, соответствующую таблице замещения tab[i], применяемой к данным x, можно записать в следующем виде:

y=tab[x].

В этом случае защита маскированием требует спонтанного генерирования случайно маскированных таблиц. Так, маскированную нелинейную операцию, соответствующую маскированной таблице замещения tab'[i], применяемой к данным x, маскированной случайной маской ml, можно записать в следующем виде:

y=tab'[x⊕m1]=y⊕m2

Чтобы иметь возможность демаскировать полученные таким образом данные y', применяют сохранение в памяти маскированных таблиц. Способы защиты такого типа были предложены для алгоритма шифрования DES в документе авторов Louis Goubin и Jacques Patarin "DES and Differential Power Analysis - The "Duplication" Methode" в Cetin Kaya Koc and Christof Paar, editors, Proceedings of CHES'99, том 1717 "Lectures Notes in Computer Science", стр.158-172, Springer-Verlag, 2000, а также в патенте FR 2802741 «Устройство для применения алгоритма поблочного шифрования с повторением раундов».

Однако такое решение может оказаться чрезвычайно дорогим с учетом необходимого объема памяти, в частности, когда немаскированная таблица замещения имеет относительно большой размер.

Например, нелинейная операция AES может быть осуществлена с использованием таблицы замещения размером 256 байт. Одновременное шифрование 16 байт сообщения требует наличия в памяти 16 маскированных таблиц замещения на 256 байт каждая. Следовательно, размер памяти, необходимый для маскирования осуществляемой таким образом нелинейной операции, должен составлять 4 кбайт.

Таким образом, недостатком данного типа защиты является то, что он требует значительного объема памяти.

Известен также документ "Provably Secure Masking of AES", Johannes Blomer, Guajardo Merchan, Volker Krummel, опубликованный 30.04.04, и документ "Secure and Efficient Masking of AES - A Mission Impossible" версия 1, E.Oswald, Stephan Mangard, Norbert Pramstaller, 4 июня 2004 г., в котором предложено осуществлять нелинейную операцию алгоритма AES в конечном поле GF(4).

В этой последней статье предложен метод маскирования операций, в котором нелинейную операцию на промежуточных маскированных данных преобразуют в линейную операцию путем транспонирования из одного конечного поля ($GF(2^8)$) в другое (GF(4)).

Вместе с тем, такой способ маскирования AES предполагает осуществлять нелинейные операции в GF(4) и, следовательно, обрабатывать биты попарно.

Как правило, легче производить эффективные операции на блоках бит размером, по существу, равном числу бит, обрабатываемых одновременно используемым микропроцессором, чем операции, осуществляемые на блоках бит размером, существенно отличающимся от числа бит, обрабатываемых одновременно микропроцессором.

Таким образом, эффективную реализацию алгоритма шифрования с обработкой

бит попарно сложно производить на 8, 16, 32 и даже 64-разрядных микропроцессорах.

Настоящее изобретение призвано предложить решение для устранения этих недостатков.

Первым объектом настоящего изобретения является способ выполнения криптографического вычисления в электронном компоненте согласно определенному криптографическому алгоритму, включающему, по меньшей мере, одну точно определенную нелинейную операцию на блоках данных из k бит, при этом k является целым числом, превышающим 2, при этом способ содержит следующие этапы:

- генерируют несколько промежуточных маскированных блоков данных из j бит на основании исходного блока данных из k бит, при этом j является целым числом, меньшим k;

- производят нелинейную операцию, по меньшей мере, на одном из j-битовых маскированных промежуточных блоков данных при помощи таблицы замещения с 2^j входами, получая j-битовый измененный блок данных;

- измененный j-битовый блок данных объединяют, по меньшей мере, с некоторыми из указанных j-битовых маскированных промежуточных блоков данных в один итоговый k-битовый блок, соответствующий исходному k-битовому блоку данных, через преобразование, включающее указанную точно определенную нелинейную операцию.

Таким образом, можно реализовать нелинейную операцию, точно определенную для блоков данных размером k, на блоках данных размером j, генерируя несколько промежуточных блоков данных размером j, при этом размер выражается в битах, на основании блока данных размером k. После обработки промежуточных блоков данных и, в частности, применения нелинейной операции объединяют полученные таким образом блоки данных, чтобы получить блок данных размером k, соответствующий преобразованию исходного блока данных. Это преобразование включает точно определенную нелинейную операцию. Оно может также включать другие операции алгоритма. Действительно, может бть более предпочтительным осуществлять другие операции алгоритма на j-битовых блоках данных, чем на k-битовых блоках данных, прежде чем объединять эти блоки, чтобы снова получить k-битовый блок данных.

Этап генерирования нескольких промежуточных блоков данных на основании исходного блока данных может содержать несколько этапов, что будет подробнее описано ниже. Так, предпочтительно обрабатывают блоки данных размером, меньшим, чем исходный блок данных, предназначенный для шифрования при помощи алгоритма. Следовательно, точно определенная нелинейная операция, включенная в алгоритм шифрования, может применяться для промежуточных блоков данных меньшего размера, и поэтому таблица замещения, соответствующая нелинейной операции, применяемой для промежуточных блоков данных, имеет размер, строго меньший, чем размер таблицы замещения, которая соответствовала бы точно определенной нелинейной операции алгоритма, применяемой для исходного блока данных.

Благодаря этому можно защищать при помощи маскирования криптографические вычисления алгоритма шифрования, содержащего нелинейную операцию, в частности, в случае, когда эта нелинейная операция соответствует таблице замещения относительно большого размера.

Этап маскирования предпочтительно можно осуществлять либо на исходных блоках данных перед этапом генерирования блоков промежуточных данных, либо на

блоках промежуточных данных.

В варианте выполнения настоящего изобретения этап генерирования содержит операцию разложения (T), состоящую в разложении k-битовых блоков данных (a) на j-битовые блоки данных (b, c), и этап объединения содержит обратную операцию разложения (T^{-1}), состоящую в создании k-битового блока данных (a') на основании j-битовых блоков данных (B, C).

В варианте выполнения настоящего изобретения криптографическое вычисление алгоритма дополнительно включает, по меньшей мере, одну линейную операцию, и маскированную линейную операцию осуществляют перед операцией разложения T или после обратной операции разложения T^{-1}. В этом случае, если алгоритм шифрования содержит линейные операции перед нелинейной операцией, операцию разложения T осуществляют после линейных операций. Если алгоритм шифрования содержит линейные операции после нелинейной операции, обратную операцию разложения T^{-1} осуществляют после линейных операций.

В другом варианте выполнения, если алгоритм шифрования включает линейные операции, то эти маскированные линейные операции осуществляют после операции разложения T и перед обратной операцией разложения T^{-1}. Таким образом, если такой алгоритм шифрования является алгоритмом с повторением раундов, при этом каждый из раундов содержит, по меньшей мере, одну линейную операцию и, по меньшей мере, одну нелинейную операцию, каждый раунд осуществляют после операции разложения T и перед обратной операцией разложения T^{-1}.

Можно также применять операцию разложения T в начале алгоритма шифрования и обратную операцию разложения T^{-1} - в конце алгоритма шифрования. Таким образом, если такой алгоритм шифрования является алгоритмом с повторением раундов, все операции всех раундов алгоритма осуществляют, обрабатывая блоки строго меньшего размера, чем исходный блок данных.

На фиг.2 показан такой вариант выполнения, применяемый для алгоритма типа AES. Так, предназначенный для шифрования блок данных 201 размером k разбивают при помощи операции разложения T на несколько предназначенных для шифрования промежуточных блоков данных (202) размером j. В этом случае для промежуточных блоков данных размером j применяют операции, защищенные маскированием (203), эквивалентные операциям алгоритма AES, точно определенным для блоков данных размером k. Получают зашифрованные промежуточные блоки данных 204. Затем применяют обратную операцию разложения T^{-1} для получения зашифрованного блока данных 206 размером k. Таким образом, этот вариант выполнения требует изменения операций алгоритма, чтобы их можно было применять для промежуточных блоков данных.

Этап генерирования маскированных промежуточных блоков данных может содержать маскированные сложения и маскированные умножения, которые осуществляют согласно алгоритму маскированного умножения, использующему на входе маскированные блоки данных размером j и случайную маску и выдающему маскированное произведение двух немаскированных блоков данных.

Этап объединения промежуточных блоков данных и измененного блока данных может также содержать маскированные сложения и маскированные умножения, которые осуществляют согласно алгоритму маскированного умножения, использующему на входе маскированные блоки данных размером j и выдающему маскированное произведение двух немаскированных блоков данных.

Как детально поясняется далее, когда исходный блок данных имеет размер в один байт и когда можно осуществлять точно определенную нелинейную операцию при помощи таблицы замещения размером, равным 256 байт, как в алгоритме типа AES, то предпочтительно можно применять нелинейную операцию для промежуточного блока данных, используя таблицу замещения размером 8 байт.

В варианте выполнения настоящего изобретения нелинейная операция является биективной и для не нулевых элементов соответствует мультипликативной инверсии в конечном поле. Предпочтительно условно эта операция приводит элемент 0 в соответствие с элементом 0.

В варианте выполнения настоящего изобретения алгоритм шифрования, использующий на входе сообщение, содержащее определенное число исходных блоков размером k, последовательно обрабатывает каждый из исходных блоков данных. Таким образом, генерируют и вводят в память таблицу замещения для каждого исходного блока данных, затем один за другим обрабатывают каждый блок данных.

В другом варианте выполнения настоящего изобретения алгоритм шифрования обрабатывает одновременно все блоки данных входящего сообщения. Таким образом, генерируют и вводят в память одновременно таблицы замещения для каждого из исходных блоков данных сообщения, затем одновременно обрабатывают указанные блоки данных сообщения. Этот тип одновременной обработки требует области памяти большего размера, чем последовательная обработка блоков данных. С этой точки зрения, как будет подробно пояснено ниже, оказывается наиболее предпочтительным уменьшить таблицу замещения, соответствующую нелинейной операции, как предлагается настоящим изобретением.

В варианте выполнения настоящего изобретения исходный блок данных обрабатывают, используя каждый раз при осуществлении нелинейной операции алгоритма шифрования таблицу замещения, генерированную в начале алгоритма шифрования. Таким образом, одну и ту же таблицу замещения используют во время всей обработки исходного блока данных при помощи алгоритма шифрования.

Можно также генерировать и сохранять в памяти таблицу замещения перед каждой маскированной нелинейной операцией. В этом варианте выполнения перед каждым осуществлением нелинейной операции используют новую таблицу замещения.

Такие алгоритмы классически содержат несколько раундов, каждый из которых содержит линейные операции и, по меньшей мере, одну нелинейную операцию. Наиболее часто встречающиеся атаки на выполнение вычислений в таких алгоритмах основаны на обнаружении утечки информации во время выполнения первых раундов и последних раундов, так как именно эти раунды обрабатывают блоки данных, близкие к предназначенным для шифрования блокам данных и к зашифрованным блокам данных. Таким образом, для защиты таких алгоритмов от атак можно маскировать блоки данных, обрабатываемые в первом или первых раундах и/или в последнем или последних раундах. Следовательно, предпочтительно генерировать и вводить в память таблицы замещения, по меньшей мере, для первого раунда и, по меньшей мере, для последнего раунда.

Вторым объектом настоящего изобретения является электронный компонент выполнения криптографического вычисления согласно определенному криптографическому алгоритму, включающему, по меньшей мере, одну точно определенную нелинейную операцию на k-битовых блоках данных, при этом k является целым числом больше 2, при этом компонент содержит:

- средства генерирования нескольких j-битовых маскированных блоков на

основании исходного k-битового блока данных, при этом j является целым числом меньше k;

- средства применения нелинейной операции, по меньшей мере, для одного из j-битовых маскированных блоков при помощи таблицы замещения с 2^j входами с последующим получением измененного j-битового блока;

- средства объединения измененного j-битового блока и, по меньшей мере, некоторых из указанных j-битовых маскированных блоков в один итоговый k-битовый блок, соответствующий исходному k-битовому блоку данных, через преобразование, включающее указанную точно определенную нелинейную операцию.

Благодаря этому можно защитить от атаки путем обнаружения утечек информации электронный компонент, использующий алгоритм шифрования типа AES, содержащий нелинейную операцию, при помощи манипулирования маскированными блоками данных, и используя маскированные таблицы замещения размером, равным 8 байт, вместо маскированных таблиц замещения размером, равным 256 байт.

Следовательно, такой способ защиты алгоритма шифрования требует относительно небольшого размера памяти. Он может применяться в электронных компонентах шифрования с относительно небольшой областью памяти.

Кроме того, в варианте выполнения настоящего изобретения обрабатывают промежуточные блоки данных размером в 4 бит. Такой размер позволяет более легко и эффективно осуществлять операции, чем при обработке блоков данных меньшего размера.

Другие аспекты, задачи и преимущества настоящего изобретения будут более очевидны из нижеследующего описания одного из вариантов его выполнения.

Изобретение будет более понятно при рассмотрении прилагаемых чертежей, на которых:

фиг.1 иллюстрирует способ защиты алгоритма шифрования согласно варианту выполнения настоящего изобретения;

фиг.2 иллюстрирует способ защиты алгоритма шифрования типа AES согласно варианту выполнения настоящего изобретения.

В варианте выполнения настоящего изобретения используют хорошо известные математические свойства конечных полей (или полей Галуа, Galois Fields (GF)), чтобы на основании исходного блока данных размером k генерировать несколько блоков данных размером строго меньше k. В описании представлен вариант выполнения настоящего изобретения для защиты путем маскирования алгоритма шифрования типа AES, использующего на входе сообщение, состоящее из 16 байт, при этом каждый из байт обрабатывают путем повторения раундов, каждый из которых содержит последовательность линейных операций и, по меньшей мере, одну нелинейную операцию. Изобретение касается и других типов алгоритма шифрования, содержащих, по меньшей мере, одну нелинейную операцию, независимо от того, осуществляется ли он с повторением раундов или нет.

В алгоритме типа AES нелинейная операция соответствует мультипликативной инверсии для ненулевых элементов в конечном поле $GF(2^8)$. Таким образом, чтобы на основании одного блока данных генерировать несколько промежуточных блоков данных строго меньшего размера, в варианте выполнения настоящего изобретения предлагается транспонировать нелинейную операцию из поля $GF(2^8)$ на поле $GF((2^4)^2)$.

Известен документ авторов Tsing-Fu Ling, Chih-Pin Su, Chih-Tsun Huang и Cheng-Wen Wu "A high-throuhput low-cost AES cipher chip", а также документ Vincent Rijmen "Efficient implementation of the Rijndael S-box" и документ Atri Rudra, Pradeep K.Dubey, Charanjit

S.Jutia, Vijay Kurmar, Josyula R.Rao и Pankaj Rohatgi "Efficient Rijndael Encryption Implementation with Composite Field Arithmetic", затем документ R.W.Ward и T.C.A.Molteno "Efficient hardware calculation of inversion in $GF(2^8)$", в которых показывается, как нелинейная операция алгоритма AES может быть транспонирована в сложное поле $GF((2^4)^2)$ и как такое транспонирование позволяет улучшить эффективность алгоритма шифрования.

Таким образом, классически алгоритм AES использует на входе сообщение, состоящее из 16 байт. Для каждого из этих байт применяется несколько раундов, при этом каждый раунд использует подключ, производный от главного секретного ключа. Как правило, AES содержит следующие операции:

- нелинейную операцию, классически обозначаемую SubBytes, соответствующую 8-битовой таблице замещения, которая применяется для каждого из 16 байт входящего сообщения;
- линейную операцию добавления единицы или исключения, классически обозначаемую AddRoundKey, применяемую между 16 байт подключа и 16 байт входящего сообщения;
- линейную операцию, классически обозначаемую ShiftRows, соответствующую перемещению, применяемому для 16 байт;
- линейную операцию, классически обозначаемую MixColumns, применяемую для 16 байт.

Операция SubBytes алгоритма AES содержит нелинейную операцию, которая в конечном поле $GF(2^8)$ для ненулевых элементов соответствует мультипликативной инверсии. Такую операцию можно транспонировать в сложное поле $GF((2^4)^2)$ таким образом, чтобы обрабатывать блоки данных размером 4 бит перед реализацией обратной операции преобразования T^{-1} для других операций обработки байт.

Выбирают отображение $GF(2^4)$ и следующий полином поля:

$P(x)=x^2+Dx+E$;

при этом D и E являются элементами конечного поля $GF(2^4)$, таким образом, можно записать:

$GF(2^8)\cong GF(2^4)[x]/P(x)$

На фиг.1 показан вариант выполнения настоящего изобретения.

Конструируют операцию 102 разложения T, которая записывается следующим образом:

$T(a)=bx+c$;

где a является элементом $GF(2^8)$, $(bx+c)$ является элементом $GF((2^4)^2)$, a b и c являются элементами конечного поля $GF(2^4)$.

На основании блока данных размером в один байт эта операция разложения T генерирует промежуточные блоки данных b и c размером, равным 4 бит.

Эту операцию разложения можно осуществить в виде умножения с матрицей 8×8.

После этой операции разложения обработанные блоки данных являются промежуточными блоками данных размером 4 бит, строго меньшим размера исходного 8-битового блока данных.

Такая операция разложения T позволяет уменьшить размер обработанных блоков данных таким образом, чтобы уменьшить размер таблицы замещения, соответствующей первой нелинейной операции. Поэтому ее осуществляют перед осуществлением нелинейной операции. Вместе с тем, ее можно осуществлять на различных этапах алгоритма шифрования. Так, для каждого из 16 байт входящего

сообщения эту операцию разложения можно осуществлять либо в начале алгоритма, либо в начале каждого из раундов алгоритма, либо перед каждой реализацией нелинейной операции.

Настоящее изобретение охватывает варианты выполнения, независимо от этапа алгоритма, на котором реализуют эту операцию разложения.

После осуществления нелинейной операции с использованием таблицы замещения можно реализовать обратную операцию разложения T^{-1} для возвращения в конечное поле $GF(2^8)$ и новой обработки байт.

Настоящее изобретение охватывает варианты выполнения независимо от этапа алгоритма, на котором реализуют эту обратную операцию разложения.

В зависимости от этапов, на которых реализуют операцию разложения T и обратную операцию разложения T^{-1}, к конечному полю $GF((2^4)^2)$ адаптируют другие операции алгоритма, определенные в $GF(2^8)$.

Таким образом, если операцию разложения T осуществляют перед нелинейной операцией и если операцию обратного разложения T^{-1} осуществляют после нелинейной операции, то нет необходимости адаптировать другие операции алгоритма к конечному полю $GF((2^4)^2)$. Зато в случае, когда операцию разложения осуществляют в начале алоритма, а операцию обратного разложения T^{-1} - в конце алгоритма шифрования, все операции алгоритма приводят в соответствие с конечным полем

$GF((2^4)^2)$. В этом случае операции алгоритма шифрования заменяют их эквивалентами в конечном поле $GF((2^4)^2)$.

В варианте выполнения настоящего изобретения производят маскирование 103, по меньшей мере, промежуточных блоков данных, обработанных в момент нелинейной операции, по меньшей мере, для первого раунда и последнего раунда алгоритма AES, что будет подробнее показано далее. Изобретение охватывает также вариант выполнения, в котором промежуточные блоки, обработанные в момент нелинейной операции, маскируют для всех раундов алгоритма, а также вариант выполнения, в котором маскируют все или часть промежуточных блоков, обработанных во время исполнения алгоритма, независимо от того, являются ли эти блоки данных элементами $GF(2^8)$ или элементами $GF((2^4)^2)$.

В варианте выполнения настоящего изобретения нелинейная операция представляет собой мультипликативную инверсию в $GF(2^4)$ для ненулевых элементов.

В конечном поле $GF((2^4)^2)$ инверсию элемента (bx+c) можно записать в следующем виде:

$$(bx+c)^{-1}=b\Delta^{-1}x+(c+bD)\Delta^{-1};$$

где $\Delta=b^2E+bcD+c^2$.

Таким образом, первая нелинейная операция мультипликативной инверсии конечного поля $GF(2^8)$, транспонированная в конечное поле $GF((2^4)^2)$, соответствует умножениям, сложениям и нелинейной операции мультипликативной инверсии. Эта нелинейная операция в конечном поле $GF((2^4)2)$ может быть осуществлена при помощи таблицы замещения 4 бит на 4 бит, занимающей 8 байт.

Как было указано выше, требуется маскировать обработанные таким образом блоки данных. Маскирование сложений хорошо известно.

Чтобы маскировать умножения, вводят алгоритм 104 маскированного умножения M(x, y, z). В дальнейшем элементы m1, m2, m3, n, n', n1, n2 являются случайными

масками. Такой алгоритм на входе использует:

- маскированные элементы $GF(2^4)$, $(b \oplus m1)$ и $(c \oplus m2)$;
- маску m3.

Такой алгоритм на выходе выдает маскированное произведение в виде: $bc \oplus m3$.

Можно записать следующие этапы:

$u=(b \oplus m1)(c \oplus m2)=bc \oplus bm2 \oplus cm1 \oplus m1m2$

$v=(c \oplus m2)m1=cm1 \oplus m1m2$

$w=(b \oplus m1)m2=bm2 \oplus m1m2$

$M(b \oplus m1, c \oplus m2, m3)=u \oplus v \oplus wm1m2 \oplus m3=bc \oplus m3$

В варианте выполнения настоящего изобретения маскируют первые промежуточные блоки данных. Таким образом, получают первое маскированное разложение, которое записывается в следующем виде:

$(b \oplus m1)x+(c \oplus m2)$.

Затем используют алгоритм маскированного умножения для вычисления маскированного умножения в следующем виде:

$M(b \oplus m1, \oplus m2, m3)=bc \oplus m3$.

Затем на этапе 105 точно так же вычисляют следующие произведения для получения маскированного блока данных Δ:

$p=(bc \oplus m3)D=bcD \oplus m3D$;

$q=(b \oplus m1)^2E=b^2E \oplus m1^2E$;

$r=(c \oplus m2)^2=c^2 \oplus m2^2$.

В результате получают следующее уравнение:

$p \oplus q \oplus r \oplus m1^2E \oplus m2^2 \oplus m3D \oplus n=\Delta \oplus n$.

Таким образом, в конечном поле $GF(2^4)$ путем сложения и умножения получают маскированный блок данных $\Delta \oplus n$, к которому применяют нелинейную операцию мультипликативной инверсии.

В варианте выполнения настоящего изобретения эту нелинейную операцию осуществляют на маскированном блоке данных, используя на этапе 106 таблицу замещения tab(x), проверяя следующее уравнение:

$tab[x \oplus n]=x^{-1} \oplus n'$.

Таким образом, эта маскированная таблица замещения позволяет получить инверсию маскированного блока данных, то есть $\Delta^{-1} \oplus n'$.

Применяя алгоритм маскированного умножения, вычисляют:

- $M(b \oplus m1, \Delta^{-1} \oplus n', n1)=b\Delta^{-1} \oplus n1=B$;
- $M(bD+c) \oplus (m1D+m2), \Delta^{-1} \oplus n', n2)=(bD+c), \Delta^{-1} \oplus n2=C$.

Таким образом, путем объединения 108 маскированных блоков данных $b \oplus m1$, $c \oplus m2$ и $\Delta^{-1} \oplus n'$ получают $Bx+C$.

Таким образом, получают маскированное значение $(bx+c)^{-1}$, причем не обрабатывая для реализации этой нелинейной операции немаскированные блоки данных b, c и Δ.

Затем выполняют обратную операцию 109 разложения T^{-1} для получения элемента 110:

$a'=T^{-1}(Bx+C)$.

В варианте выполнения настоящего изобретения определяют D, равное 1, чтобы уменьшить число производимых умножений.

Предпочтительно нелинейную операцию транспонируют из конечного поля $GF(2^8)$ в конечное поле $GF((2^4)^2)$, в котором нелинейная операция соответствует комбинации сложений, умножений и эквивалентной нелинейной операции в поле с меньшим числом элементов.

Таким образом, можно защитить электронные компоненты от атак путем обнаружения утечек информации во время исполнения криптографического вычисления, содержащего нелинейную операцию, осуществляя операции, эквивалентные этой нелинейной операции, на блоках данных меньшего размера, что позволяет реализовать эти вычисления на маскированных блоках данных, так как соответствующая таблица замещения имеет меньший размер.

Формула изобретения

1. Способ выполнения криптографического расчета в электронном компоненте согласно определенному криптографическому алгоритму, включающему, по меньшей мере, одну точно определенную нелинейную операцию на блоках данных из k бит, при этом k является целым числом, превышающим 2, при этом способ содержит следующие этапы:

генерируют несколько промежуточных маскированных блоков даных из j бит ($b \oplus m$, $c \oplus m2$, $\Delta \oplus n$) на основании исходного блока данных из k бит, при этом j является целым числом, меньшим k;

производят нелинейную операцию S, соответствующую указанной точно определенной нелинейной операции, по меньшей мере, на одном j-битовом маскированном промежуточном блоке данных ($\Delta \oplus n$) при помощи таблицы замещения (106) с 2^j входами, получая j-битовый измененный блок данных ($S(\Delta) \oplus n$);

измененный j-битовый блок данных объединяют, по меньшей мере, с некоторыми из указанных j-битовых маскированных промежуточных блоков данных в один итоговый k-битовый блок (a'), соответствующий исходному k-битовому блоку данных, через преобразование, включающее указанную точно определенную нелинейную операцию.

2. Способ по п.1, в котором исходный блок данных маскируют перед этапом генерирования промежуточных блоков данных.

3. Способ по п.1, в котором промежуточные блоки данных маскируют (103) перед применением нелинейной операции.

4. Способ по п.1, в котором этап генерирования содержит операцию разложения (T), состоящую в разложении k-битового блока данных на j-битовые блоки данных (b, c), и этап объединения содержит обратную операцию разложения (T^{-1}), состоящую в составлении k-битового блока данных на основании j-битовых блоков данных (B, C) в k-битовый блок данных (a').

5. Способ по п.4, в котором криптографическое вычисление алгоритма дополнительно включает, по меньшей мере, одну линейную операцию, и согласно которому маскированную линейную операцию осуществляют перед операцией разложения (T) или после обратной операции разложения (T^{-1}).

6. Способ по п.4, в котором криптографическое вычисление алгоритма дополнительно включает линейную операцию, и согласно которому указанную маскированную линейную операцию осуществляют после операции разложения (T) и перед обратной операцией разложения (T^{-1}).

7. Способ по п.4, в котором операцию разложения (T) применяют в начале

криптографического алгоритма и обратную операцию разложения (T^{-1}) - в конце криптографического алгоритма.

8. Способ по п.1, в котором этап генерирования маскированных промежуточных блоков данных содержит маскированные сложения и маскированные умножения, которые осуществляют согласно алгоритму маскированного умножения, использующему на входе маскированные блоки данных размером j и случайную маску и выдающему маскированное произведение двух не маскированных блоков данных.

9. Способ по п.1, в котором этап объединения промежуточных блоков данных и измененного блока данных содержит маскированные сложения и маскированные умножения, которые осуществляют согласно алгоритму маскирванного умножения, использующему на входе маскированные блоки данных размером j и случайную маску и выдающему маскированное произведение двух не маскированных блоков данных.

10. Способ по п.1, в котором блок данных имеет размер в 1 байт, и таблица замещения имеет размер в 8 байт, и согласно которому нелинейная операция является биективной и для ненулевых элементов является мультипликативной инверсией в конечном поле.

11. Способ по п.1, в котором блок данных маскируют, реализуя единицу или исключение между блоком данных и случайной маской.

12. Способ по п.1, в котором криптографический алгоритм использует на входе сообщение, содержащее определенное число исходных блоков размером k, и согласно которому генерируют и вводят в память таблицу замещения последовательно для каждого исходного блока данных указанного сообщения, затем один за другим обрабатывают каждый блок данных.

13. Способ по п.12, в котором таблицы замещения генерируют и вводят в память одновременно для каждого из исходных блоков данных сообщения, затем одновременно обрабатывают указанные блоки данных сообщения.

14. Способ по п.12, в котором блок данных шифруют, используя для всех операций криптографического алгоритма таблицу замещения, генерированную в начале криптографического алгоритма.

15. Способ по п.12, в котором таблицу замещения генерируют перед каждой маскированной нелинейной операцией.

16. Способ по п.12, в котором алгоритм содержит определенное число раундов и согласно которому таблицу замещения генерируют и вводят в память, по меньшей мере, для первого раунда и, по меньшей мере, для последнего раунда.

17. Способ по п.1, в котором криптографический алгоритм является алгоритмом AES.

18. Электронный компонент, предназначенный для криптографического вычисления согласно определенному криптографическому алгоритму, включающему, по меньшей мере, одну точно определенную нелинейную операцию на k-битовых блоках данных, при этом к является целым числом больше 2, при этом компонент содержит

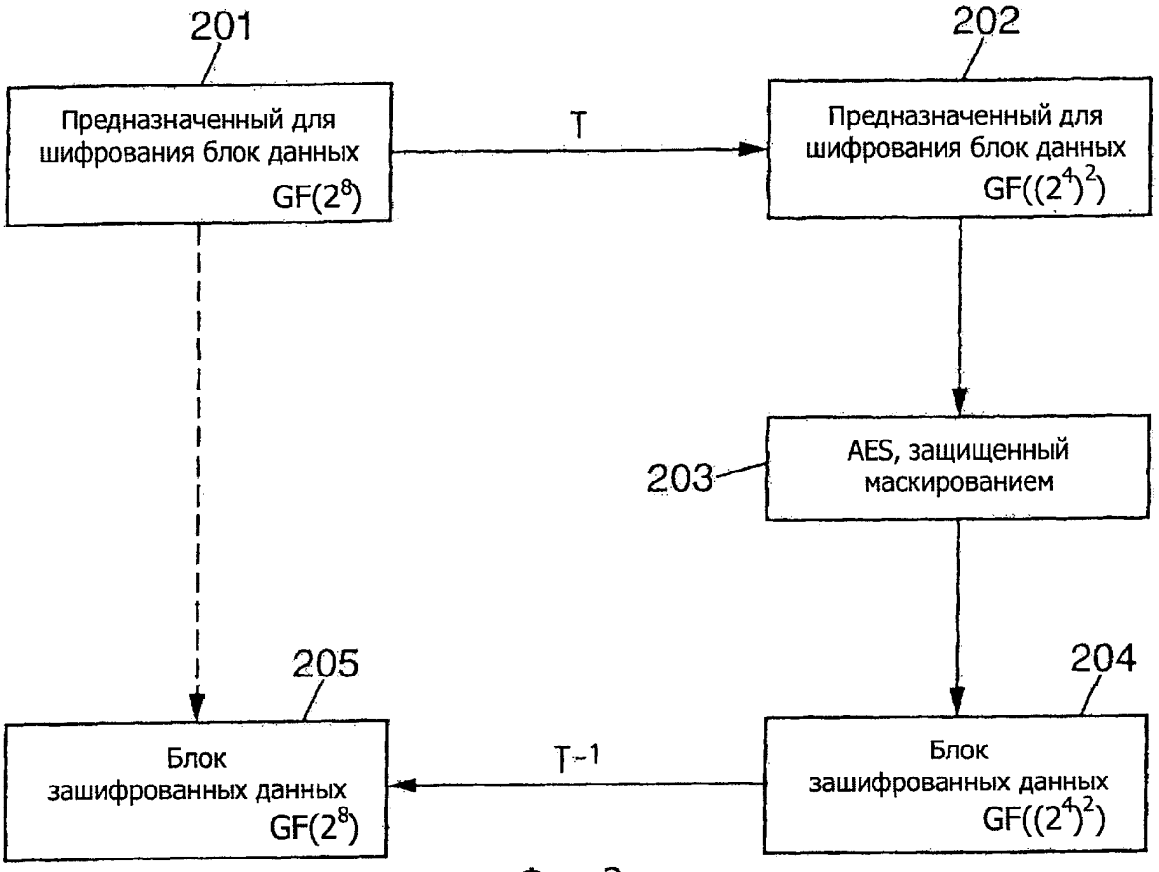
средства генерирования нескольких j-битовых маскированных блоков на основании исходного k-битового блока данных, при этом j меньше k;

средства применения нелинейной операции S, соответствующей указанной точно определенной нелинейной операции, по меньшей мере, для одного из j-битовых маскированных блоков при помощи таблицы замещения с 2^j входами с последующим получением измененного j-битового блока;

средства объединения измененного j-битового блока и, по меньшей мере, некоторх

из указанных j-битовых маскированных блоков в один итоговый k-битовый блок, соответствующий исходному k-битовому блоку данных, через преобразование, включающее указанную точно определенную нелинейную операцию.

19. Электронный компонент по п.18, в котором криптографический алгоритм является алгоритмом AES.


201
Предназначенный для шифрования блок данных
GF(2^8)
T
202
Предназначенный для шифрования блок данных
GF((2^4)^2)
203
AES, защищенный маскированием
204
Блок зашифрованных данных
GF((2^4)^2)
T-1
205
Блок зашифрованных данных
GF(2^8)

Фиг. 2