(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 1795471 B
(45)授权公告日 2010.10.13

(21)申请号 200480014723.1
(22)申请日 2004.05.27
(30)优先权数据
0953/03 2003.05.28 CH
(85)PCT申请进入国家阶段日
2005.11.28
(86)PCT申请的申请数据
PCT/IB2004/050794 2004.05.27
(87)PCT申请的公布数据
WO2004/107283 FR 2004.12.09
(73)专利权人 纳格拉影像股份有限公司
地址 瑞士舍索－苏尔－洛桑
(72)发明人 亨利·库德尔斯基 瑟奇·高梅恩
(74)专利代理机构 中国国际贸易促进委员会专利商标事务所 11038
代理人 康建忠
(51)Int.Cl.
G07F 7/10(2006.01)
(56)对比文件
FR 2829645 A1,2003.03.14, 说明书第5页第9行至第12页第18行,说明书附图1～3.
FR 2829645 A1,2003.03.14, 说明书第5页第9行至第12页第18行,说明书附图1～3.
US 5177790 A,1993.01.05, 说明书第5栏第16行至第16栏第39行,说明书附图1、2.
审查员 张田勇

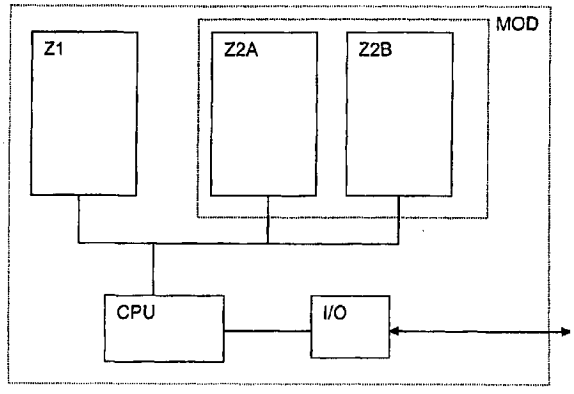
权利要求书 1 页 说明书 5 页 附图 1 页

(54)发明名称

安全性密钥生成方法

(57)摘要

本发明的目的是提供一种在例如智能卡的安全装置的第二存储器区域的内容已被第三方读取之后恢复其安全性的方法。通过一种使用安全模块来生成安全性密钥的方法而达到此目的,该安全模块包括中央处理单元、第一条件访问存储器区域和包括用户程序的全部或一部分的至少一个第二存储器区域,其特征在于,所述方法包括以下步骤:读出第二存储器区域的全部或一部分,基于第二区段区域中的数据的全部或者一部分和在第一存储器区域中存储的秘密信息中的至少一项来生成至少一个根密钥。



CN 1795471 B

1. 一种由安全模块 (MOD) 所实现的用于生成根密钥的方法，该安全模块包括中央单元 (CPU)、第一存储器区段 (Z1) 和至少一个第二存储器区段 (Z2)，所述中央单元 (CPU) 通过一个系统程序访问所述第一存储器区段 (Z1)，并访问所述第二存储器区段 (Z2)，所述第二存储器区段包括一个用户程序的全部或者一部分以及数据 (DTA)，所述用户程序只能访问所述第二存储器区段 (Z2)，其中该方法包括以下步骤：

- 读出第二存储器区段 (Z2) 的全部或者一部分，

- 基于第二存储器区段 (Z2) 中的数据 (DTA) 的全部或者一部分和在第一存储器区段 (Z1) 里存储的至少一项秘密信息 (MK2，RTN) 来生成至少一个根密钥 (RK)。

2. 根据权利要求 1 的方法，其中，该秘密信息是工厂密钥 (MK2)。

3. 根据权利要求 1 的方法，其中，该秘密信息是描述生成根密钥 (RK) 的数据 (DTA) 的使用的算法 (RTN)。

4. 根据权利要求 1 的方法，其中，所述方法包括：计算至少一项控制信息 (H)，该控制信息表示第二区段 (Z2) 的数据 (DTA) 的全部或一部分，所述控制信息 (H) 被用于生成根密钥 (RK)。

5. 根据权利要求 4 的方法，其中，所述控制信息 (H) 是在第二存储器区段 (Z2) 的数据的全部或一部分上执行的被称为单向的且无碰撞的函数 (Hash) 的结果。

6. 根据权利要求 1 的方法，其中，所述第二区段 (Z2) 还包括描述部分 (DES)，该描述部分包括确定控制信息 (H) 的构成的存储器区段的位置。

7. 根据权利要求 6 的方法，其中，所述描述部分 (DES) 包括用于和部分控制信息 (H1，H2 H2...Hn) 相对应的用户存储器区段 (Z2B) 的每个部分 (PA，PB，PC) 的多条位置信息。

8. 根据权利要求 2 的方法，其中，所述工厂密钥 (MK2) 是对称类型的。

9. 根据权利要求 2 的方法，其中，所述第二区段 (Z2) 包括验证区段 (Z2A) 和用户区段 (Z2B)，在验证区段 (Z2A) 中所包括的程序负责用户区段 (Z2B) 中的数据的验证，第一区段的程序系统把必要的数据从所述第一区段向验证区段 (Z2A) 进行传送。

10. 根据权利要求 9 的方法，其中，所述工厂密钥 (MK2) 被系统程序从第一区段 (Z1) 向验证区段 (Z2A) 进行拷贝。

11. 根据权利要求 10 的方法，其中，当根密钥被生成时所述工厂密钥被删除。

12. 根据权利要求 1 的方法，其中，根密钥 (RK) 被用作传输密钥以解密来自于管理中心的消息。

安全性密钥生成方法

技术领域

[0001] 本发明涉及包括至少一个中央单元和两个存储区域的安全性模块的领域。

背景技术

[0002] 这些单元被用于实现密码系统的操作且以单片电路的形式被给出，它们在同一硅芯片上被产生或者它们被装配在一个支撑物上并被嵌入在树脂中或者由覆盖不同元件并在企图的入侵情况下担当熔丝的薄片所保护。

[0003] 这些安全处理器具有被称为自举区段(bootstrap zone)的第一存储器区段，所述区段在处理器的激活期间或者在每次初始化时被执行。这个存储器是ROM类型的，即，它是只读存储器。

[0004] 在启动程序的执行期间，此程序验证第二存储器区段，所述区段是可重写类型的，通常是EEPROM、NVRAM或者Flash类型的。

[0005] 此验证很重要，因为它用来确保在这个第二区段中的数据是有效的，即，它一定是一程序（至少在某种程度上是）。

[0006] 此验证可以用各种方式被实现，例如，印记(imprint)(CRC，Hash)的计算以及此印记与储存在同一区段中的数值的比较。

[0007] 一旦最初已被启动的主程序完成其验证，则它切换到第二区段并在一常规地址处开始用户程序的执行。

[0008] 这类处理器的特性是：当程序在第二区段中执行时，它不能自由访问第一区段的存储器。此访问或者被明确禁止或者受验证机制（例如口令）的限制。

[0009] 此限制提供重要的安全性，因为验证手段以及启动数据对用户程序来说是不可访问的。因此保护包含在第一区段中的所有数据不受任意入侵。

[0010] 此第一自举区段除了具有只读存储器(ROM)中的一部分之外，还可能包括可重写的存储器的部分，该可重写的存储器的部分受到同样的安全性条件的限制。

[0011] 当第一区段具有非常有限的大小时，验证程序的执行可以从第二区段被执行。后者被分成验证部分和用户部分。

[0012] 因此，用户程序的验证根据第一区段的数据，即，根据通常储存在所述第一区段中并允许验证第二区段的数据印记的第一密钥，而被执行。

[0013] 第二区段包含构成程序的数据和由该第一密钥所加密的签名(signature)。

[0014] 可以位于第一区段中或者位于第二区段的验证部分中的验证程序计算关于要被验证的数据的唯一印记(Hash，CRC)。

[0015] 为了验证数据被正确地证实，第二区段包含被最初储存在第一区段中的由密钥所加密的印记。此密钥被用来解密所加密的印记，并且所获得的结果与计算出的印记进行比较。

[0016] 这个密钥可以以确定的形式(ROM)或者以编程的形式(EEPROM或者Flash)位于第一区段中。在所述第二种情况中，编程在制造步骤期间或者在一个经授权的中心里被执

行，只要在此存储器位置中没有发现其它密钥，第一区段的程序就接受这种写入。

[0017] 此密钥可以是对称的类型的并且因此是秘密的，或者它可以是不对称的类型的。在此第二变形中，此密钥可以在除了第一区段之外的存储器区段中被发现，因为即使第三方发现了此密钥，所述第三方也不能识别被修改的数据集，因为他必需具有相应的私有密钥来识别所述数据。显然，此密钥被负责准备该数据更新的管理中心保持秘密。

[0018] 第二存储器区段的数据可以表示一个或者几个程序、诸如权利或者解密密钥的重要数据、或者二者的组合。

[0019] 用于发现第二区段的内容的已知类型的攻击之一是搜索安全性缺陷，比如允许对处理器进行控制的存储器溢出。一旦成功地进行控制，则第三方向外部传送第二区段的内容并且能够分析所使用的安全性机制和密钥。

[0020] 使用第二存储器区段的内容的知识，所述第三方具有用来管理对于控制此处理器的业务的各种权利和访问的密钥。

[0021] 因此，如果由管理中心管理的密钥发生改变，则此改变命令将被第二存储器区段中的密钥所加密。知道此密钥的第三方就能够解密这则消息并且也更新此新密钥的内容。

[0022] 因此，明显地，虽然可靠的机制被用于验证程序区段（第二区段）的内容，但是一旦安全性已被破坏，则管理中心发出的任何改变都无法影响安全性，因为改变手段（例如新的传输密钥）使用第三方已经拥有的密钥。他因此能够解密更新消息并且也改变其传输密钥。即使在应用中已经纠正了安全性缺口（security breach），但是破坏仍然无法被停止。

发明内容

[0023] 本发明的目的是提出一种方法，一旦第二存储器区段的内容已被第三方读取，则该方法恢复这类安全性装置的安全性。

[0024] 使用一种由安全性模块实现的用于生成安全性密钥的方法来达到此目的，此安全性模块包括中央单元、第一条件访问存储器区段和至少一个第二存储器区段，所述第二存储器区段包含用户程序的全部或一部分，其中该方法包括如下步骤：

[0025] - 读出第二存储器区段的全部或一部分，

[0026] - 基于第二区段的数据的全部或者一部分和存储在第一存储器区段中的秘密信息的至少一项来生成至少一个根密钥。

[0027] 因此，由于此新的根密钥的生成，有可能保证传输密钥的替换并且以同样的方式保证随后传输的所有密钥的替换。

[0028] 重要的是，这个根密钥决不是固定的，并且因为所述原因而必需不同于诸如工厂密钥的存储在第一存储器区段中的任何密钥。为此原因，使用管理中心所传输的新数据作为变量来生成所述根密钥。

[0029] 在第一种形式中，生成这个新密钥而不必验证第二区段的数据。如果此数据已被修改，则根密钥将只须成为错误的并且通过此密钥对传输密钥的将来的解密不会给出正确的结果。

[0030] 此根密钥因此一方面取决于第二存储器的下载或内容（或者数据），另一方面取决于储存在第三方不可访问的位置中的密钥。

[0031] 根据另一实施例，工厂密钥被储存在第一区段中的秘密程序所替换，该秘密程序根据秘密算法计算关于第二区段数据的全部或一部分的印记。根据特定的算法对第二区段数据的操作（组合、乘法、除法、EXOR 等）使得根密钥被确定。

附图说明

[0032] 根据如下的详细说明并且参考作为非限制性示例给出的附图，将会更好地理解本发明，其中：

[0033] 图 1 描述了安全性处理器装置的构造；

[0034] 图 2 示出了第二区段的划分；

[0035] 图 3 描述了用于生成根密钥的机制。

具体实施方式

[0036] 在图 1 中，模块 MOD 是安全性处理器模块。为此原因，它处理至少两个存储器区域，即，第一区段 Z1 和第二区段 Z2。第一区段包括所有或部分 ROM 存储器，并且因此是不可重写的。另外，其中也可能有一部分包括用于变量的 RAM 或 EEPROM 形式的存储器。由于特别是在第二区段中的程序的执行期间，其不是可自由访问的，因此这被称为条件访问。

[0037] 第二区段 Z2 包含处理程序和数据。这个区段包括非易失性存储器，但是有可能写入此类 EEPROM。区段 Z2 还可以包含诸如 RAM 的易失性存储器。事实上，这个区段通常不是都同一类的并且可以包括几个 ROM、RAM、EEPROM、NVRAM 和 FLASH 类型的存储器。

[0038] 在本示例中，被称为工作区段 Z2A 的区段 2 的第一部分被考虑用来执行与根密钥的生成相关的操作。

[0039] 用户区 Z2B 是包含处理程序的部分的示意图。根据实现方法，其可能包括诸如安全性密钥的变量。

[0040] 处理器 CPU 在初始化或者重置期间在第一区段 Z1 中被自动搜寻。此时第一安全性操作被执行。

[0041] 这些操作使用第一存储器区段，但是，如果有必要也使用工作区段 Z2A。由于第一区段的有限区域，消息被发送给工作区段以便例如执行印记的计算。允许计算此印记的例行程序可以在第二区段中被找到。什么也不能阻止此例行程序形成将被验证的数据的一部分。这个程序被称为系统程序。

[0042] 最初启动的初始化程序计算有关要被验证的数据的常规部分的印记。这个部分由在第二存储器区段中包含的指针所定义。用户区段 Z2B 的部分机制的说明被包含在图 2 中。

[0043] 对形成印记的数据的考虑可以在所有的或一部分的用户区段上进行。在实践中，优选地，此印记将是在程序部分上被计算的而不是在数据部分上被计算的（例如可视化权利），因为后者在用户程序的使用期间易于修改。在系统启动时初始化的印记的识别程序计算有关要被验证的数据的预确定部分的所述印记。这个部分由在第二存储器区段中包含的指针，尤其是在图 2 中的部分 DES 中包含的指针所定义。

[0044] 在本发明的范围内，此印记通过单向操作来实现，此单向操作是源集朝着目的地集的数学运算 H，其中源集的每个元素 x 用图像 (image)H(x) 表示。这些函数在它们是所谓的 Hash 函数时特别有用，例如在著作" RSA Laboratories' Frequently Asked Questions

AboutToday′ s s Cryptography，v4.0″的第 27 页中定义的内容。元素 x 可以是任意长度的，但是 H(x) 总是固定长度的字符，即，一个固定大小的串。这类函数难以反转，也就是说，知道 H(x) 一般不能发现 x。此外，它是无碰撞的 (collision free)，因为它是单射函数 (injective function)，也就是说，H(y) ＝ H(x) 必定导致 y ＝ x，类似地 H(y) ≠ H(x) 必定导致 y ≠ x。

[0045] 只要集 x 的单个数值被修改，即使其它值被修改以使第一修改生成的修改无效，再现同一控制信息 H 也被认为是不可能的。

[0046] 在图 2 中，图 1 中的用户区段 Z2B 被分成几个部分 PA、PB 和 PC。这些部分在这个示例中不是相邻的并且被不影响印记计算的部分 PI 分开。描述这些不同部分的信息被包含在也形成用户区段 Z2B 的一部分的部分 DES 中。它包含控制信息的计算中涉及的存储器位置的指示。这些指示可以是以″启动指针″和″长度″的形式的或者是以″启动指针″和″结束指针″的形式的。

[0047] 此外，具有不只一项而是几项控制信息是可能的，每项信息 H1、H2、Hn 被应用在部分 PA、PB 或 Pn 上。这允许生成不止一个根密钥而是生成几个密钥。

[0048] 在图 1 中，I/O 块说明了向模块 MOD 的外部进行通信的手段，其是使用密码函数和储存在存储器 Z2B 中的权利的不可缺少的手段。还是用这种方式，通过例如先前描述的缺陷的缺陷，数据从区段 Z2 中随机提取。

[0049] 在图 3 中，根密钥的生成被示意地表示。根据图 2 中的示例，包括部分 PA、PB 和 PC 的数据 DTA 用来利用处理器计算印记，在我们的情况中，该印记是控制信息 Hash。为了计算根密钥 RK，这个控制信息 H 和工厂密钥 MK2 通过加密模块 ENC 被用来获得所述根密钥 RK。这个秘密密钥将是对称类型的 (或者由管理中心对称地使用)，因为在相反的情况下，它在管理中心中和在模块 MOD 中不是相同的合成根密钥。

[0050] 应该注意，如果当储存的程序的一致性被验证时用户部分 Z2B 的内容已经具有建立的印记，则使用所述印记代替控制信息 H 是可能的。在这个操作中重要的因素是表示数据 DTA 的全部或者一部分的数据的使用。在一种变形中，有可能从三个八位位组中选择一个例如来识别将被工厂密钥 MK2 加密的数据。

[0051] 根据另一实施例，工厂密钥被储存在第一区段 Z1 中的密秘算法 (RTN) 替换。必要时，所述算法可以在初始化阶段期间从该第一区段拷贝到工作区段 Z2A。

[0052] 根据一个特定的方法，所述算法合并数据 DTA 的全部或者一部分以便获得取决于所述数据的唯一结果。这种组合可以实现不同的算术操作，比如乘法、Exor 等等。

[0053] 一旦已经计算出这个根密钥，则它被储存在第二区段 Z2 的存储器区段中。

[0054] 这些方法步骤的执行的位置未被识别。自举区段中的程序可以只需把工厂密钥拷贝到一个临时存储器区段中，并且被称为系统程序的根密钥生成程序可以被包含在工作区段 Z2A 中。重要的因素是所述工厂密钥在第一区段 Z1 中的存储以便在用户程序的正常执行期间使其不可访问。

[0055] 一旦根密钥已被生成，则从临时存储器中删除工厂密钥。

[0056] 根据一个实际应用，负责安全性的管理中心准备新的软件以避免已知的缺陷，例如目的是提取区段 Z2 的数据的伪造者的攻击。此新的软件被签名 (sign)，也就是说，关于该数据来计算 Hash 函数，并且用专用密钥 MK1 加密该结果。

[0057] 所有内容然后都被传输密钥加密并且以消息的形式发送给安全性模块 MOD。

[0058] 在用户区段 Z2B 中的程序处理输入数据并且借助于一个或几个系统传输密钥解密所述消息。数据然后被储存在为此目的所提供的位置中。一旦已经完成此下载，则处理器激活重启函数。这允许所有新存储的数据被验证。

[0059] 此验证一般是指储存的程序集，并且所述验证是根据上述步骤来执行的。如果具有不安全模块 MOD 的第三方的假设被考虑，则第一存储器区段 Z1 不存在（或者为空），并且处理器立即在第二区段 Z2 中启动。从管理中心接收到的新程序被第三方解密并且用户区段因此与具有双存储器区段的安全处理器的相同。

[0060] 在安全处理器的起动期间，根密钥被生成并且被用来解密新的传输密钥。伪造的模块不具有这个根密钥并且不能解密传输密钥。此时，在管理中心和安全模块之间交换的消息对伪造的模块来说不再是可访问的。如果后者尝试借助于允许其获得第二区段内容的这种类型的攻击来重新发现根密钥，则这种攻击将不再起作用，因为这个新软件的目的正好是避免这类欺骗。安全的模块在允许数据提取的攻击之前重新发现安全级别。

[0061] 因此，这种方法允许远程地改正安全性缺陷并且重置原始的安全性而无需象通常的情况那样交换所有的模块。

[0062] 如上所述，对第一区段 Z1 的访问在微处理器起动时或者在验证机制之后被执行。在上述情况期间，有可能不用激活微处理器的重置，并且通过网关请求访问第一区段。一旦通过此请求网关（通过例如口令的引导）执行进入，则程序的执行不再是可见的，因为第二区段对重新拷贝此区段的第三方来说是未知的。用这种方式启动的程序开始生成根密钥。

[0063] 条件访问存储器区段 Z1 不能提供必要的秘密数据来形成根密钥。在这种配置中，用户区段 Z2 的程序只能访问第一区段 Z1 以便读取用于计算根密钥的数据。在这些操作期间，第一区段的可视性持续时间将被限制为用于读取所必需的时间，然后将使此区段不可访问。

[0064] 根据一个实施例，工厂密钥产生一组密钥。在每次生成根密钥时，工厂密钥被停用。要被使用的密钥的选择可以用不同的方式来执行，即：

[0065] - 根据管理中心的命令，也就是说，通过定义数据 DES 中的描述符，

[0066] - 通过使用印记的 n 个最后的比特（例如 3 个比特），其允许所述比特从储存的密钥（例如 8 个密钥）中进行选择。

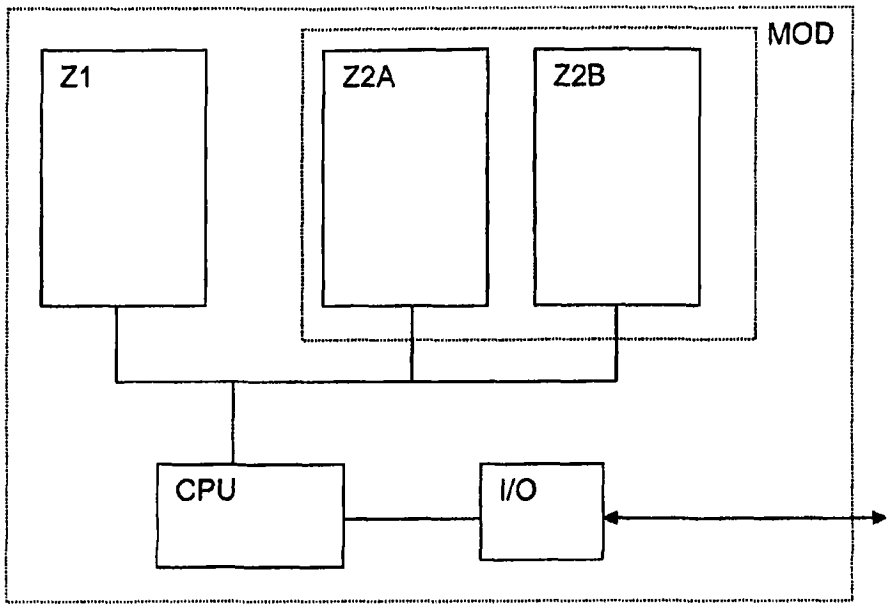


图1

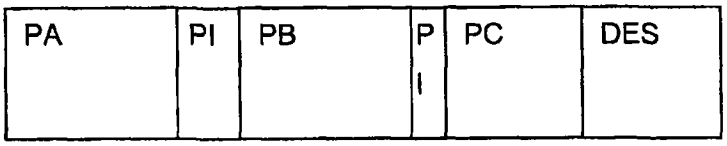


图2

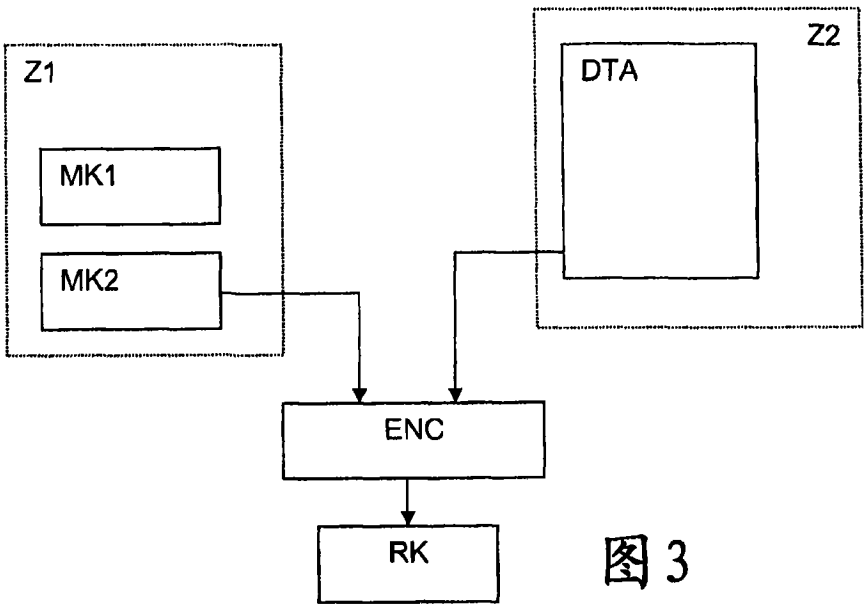


图3