[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
H04L 9/30 (2006.01)

[12] 发明专利说明书

专利号 ZL 02816198.X

[45] 授权公告日 2008年4月9日

[11] 授权公告号 CN 100380861C

[22] 申请日 2002.8.16 [21] 申请号 02816198.X
[30] 优先权
[32] 2001.8.20 [33] FR [31] 01/10,938
[86] 国际申请 PCT/FR2002/002896 2002.8.16
[87] 国际公布 WO2003/017569 法 2003.2.27
[85] 进入国家阶段日期 2004.2.19
[73] 专利权人 法国电信局
地址 法国巴黎
[72] 发明人 M·杰罗特
[56] 参考文献
EP 0666664 A1 1995.8.9
US 5966445 1999.10.12
审查员 刘剑波

[74] 专利代理机构 上海专利商标事务所有限公司
代理人 钱慰民

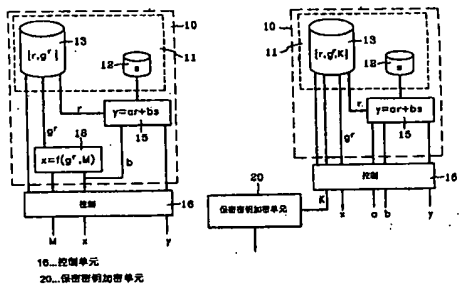
权利要求书2页 说明书7页 附图2页

[54] 发明名称

使用离散对数函数产生不对称加密系统的加密单元的方法

[57] 摘要

本发明涉及一组公开密钥加密方案，它们为了降低研发、生成并维护加密单元的成本而使用离散对数问题。实体(10)之一执行一计算，该计算最多包括少量的整数加法、加法和乘法，所述计算是组中所有方案所共有的。上述计算最好是要被所述实体执行的主要计算，而大多数其它计算可以预先被执行。特别是，所述计算属于 y = ar + bs 类型，其中 r 是随机数，s 是对于实体(10)特定的保密密钥。所述计算是用于实体验证、消息验证、数字签名和密钥交换的一组方案所共有的。



1．不对称加密系统中的一种使用整数保密密钥 s 的加密方法，其特征在于，所述保密密钥 s 与一公开密钥相关联，并且在包括在第一实体内并且独立于所述加密系统构造的一个组件（15）中给出一整数值 y，所述公开密钥包括第一元素 g 和第二元素 $v=g^s$ 或 $v=g^{-s}$，所述给出包括：

第一实体随机地选择一整数 r，根据属于进行乘法的集合 G 的元素 g^r 的函数来计算一个值 x，并把所述值 x 发送至所述第一实体外部的第二实体；

所述第二实体随机地选择两个进一步的整数运算数 a 和 b 并且把所述进一步的整数运算数发送至所述第一实体；

所述第一实体中的所述组件(15)根据 y=ar+bs、y=ar-bs、y=bs-ar 或 y=-ar-bs 计算所述整数值，并把所述整数值 y 发送至所述第二实体；以及

所述第二实体使用分别对应于计算公式 y=ar+bs、y=ar-bs、y=bs-ar 或 y=-ar-bs 的校验公式 $g^y = x^a v^b$、$g^y v^b = x^a$、$g^y x^a = v^b$ 或 $g^y x^a v^b = 1$ 进行校验。

2．如权利要求 1 所述的方法，其特征在于，所述方法还包括接收预先计算的 $\{r, x\}$ 或 $\{r, g^r\}$ 对。

3．如权利要求 1 所述的方法，其特征在于，所述进一步的运算数 a 和 b 是从控制模块接收的，所述值 x 和整数 y 被发送到该控制模块。

4．如权利要求 1 所述的方法，其特征在于，所述进一步运算数之一 a 等于 1。

5．如权利要求 4 所述的方法，其特征在于，所述进行乘法运算的集合 G 拥有一个组结构。

6．如权利要求 5 所述的方法，其特征在于，所述组件（15）这样安排，使得从值 x 和整数 y 被发送至的控制模块接收另一个进一步运算数 b，且其中获得值 x 作为元素 g^r 的函数包括应用哈希函数。

7. 如权利要求 5 所述的方法，其中所述组件（15）这样安排，使得从值 x 和整数 y 被发送至的控制模块接收另一个进一步运算数 b，且其中值 x 是元素 g^r 以及消息（M）的内容的函数，消息（M）要被包括实现加密方法的所述第一实体的装置所验证，所述控制模块包含在所述第二实体内。

8. 如权利要求 5 所述的方法，其中进一步的运算数 b 作为值 x 的函数被计算，且值 x 是元素 g^r 以及消息（M）的内容的函数：$x = f(g^r, M)$，消息（M）要被包括实现加密方法的所述第一实体的装置所验证。

9. 如权利要求 1 所述的方法，其中所述第一实体带有与第二实体进行通信的装置，值 x 和整数 y 被发送至所述第二实体，所述第二实体与另一整数保密密钥 s' 相关联，且所述方法还包括：把一公开密钥(g', v')与保密密钥 s' 相关联，所述公开密钥(g', v')包括元素 g'=g 和集合 G 的另一元素 v'，使 $v'=g^{s'}$，其中由随机数 r、所述值 x 和依赖于集合 G 的元素 v'^r 的共有密钥 K 构成三元组{r, x, K}或{r, g^r, v'^r}，所述三元组未被发送至所述第二实体。

10. 如权利要求 9 所述的方法，其特征在于，所述方法还包括接收预先计算的三元组{r, x, K}或{r, g^r, v'^r}。

使用离散对数函数产生不对称加密系统的加密单元的方法

本发明涉及加密的技术领域，尤其涉及所谓不对称的或公开密钥的加密。

在这种加密类型中，每个用户为了给定的用途而保留一对密钥，包括保密密钥和相关的公开密钥。

例如，如果正在处理专用于机密性的密钥对，则使用公开密钥对数据进行加密，而使用保密密钥对它们解密，即对这些数据进行译码。如果正在处理专用于数据真实性的密钥对，则使用保密密钥对数据进行数字签名，而使用公开密钥来校验数字签名。其他用途(实体验证、密钥交换等等)也是可行的。

与保密密钥加密不同，公开密钥加密非常有用，因为它不要求涉及的用户分享秘密，以便建立安全的通信。然而，这种安全性的优点伴随以性能的缺点，因为对于相等的资源，公开密钥加密方法(也称为“公开密钥方案”)通常比所谓的保密密钥加密方法(也称为“保密密钥方案”)慢一百或一千倍。因此，为了获得合理的计算次数，实现这些算法的电路成本通常很高。

这对于所谓 RSA 数字加密和签名方案(见 R.L. Rivest, A. Shamir 和 L.M. Adleman 撰写的“A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”，登载于“Communications of the ACM”，第 21 册，第 2 号，120－126 页，1978 年 2 月)来说尤其成立。该方案依赖于整数因式分解问题的难点：假定一个大整数(一般在以 2 为基数的表示中大于 1000 比特)等于两个或多个大小可比的质因数的乘积，不存在用于重现这些质因数的有效过程。因此，这个方案中执行的计算涉及很大的数字。它们不能在芯片卡上少于 1 秒的时间内被执行，除非后者配备了专用加密协处理器，这大大增加了它的成本。此外，由于因式分解过程的效率随时间相当快速地增长，因此需要向上修改密钥长度，损害了性能。

因此产生降低实现公开密钥方案的芯片成本的问题。

对付这个问题主要有两种方法。第一种方法在于最好(但不必要)根据因式分解以外的其他问题指定新的加密方案，这可能大大加速计算时间。该途径已被多次探索，且产生了多种结果。然而，在绝大部分情况下，或者与 RSA 相比的改进不足以设想其中的替代，或者尚未充分好地建立安全性。

第二种方法在于以大量制造芯片使得其成本能大大降低。如果国际银行组织确认该方案用于将来基于芯片的银行卡，这就是可能发生在 RSA 上的情况。然而，起初 RSA 芯片的成本太高，无论制造多少芯片，其费用仍会是可观的。

应该注意的是，许多公开密钥加密方案共同地把整数运算用作基本运算，比如模数乘法(ab(模 n))、模数除法(a/b(模 n))或者模数取幂(a^b(模 n))，其中 a、b 和 n 是整数。然而，这些运算不会完全相同。因而，每次修改加密方案时，都有必要改变执行加密计算的安全装置的程序或电路。

本发明的一个目的是通过结合上述两种方法而降低公开密钥加密单元的成本。

因此，本发明提出了一种在不对称加密系统中用于产生与整数保密密钥 s 相关联的加密单元的方法，其中所述加密单元配备了一个组件，该组件独立于加密系统而产生，并且适用于在几个整数运算数之间的组合来给出整数 y，所述几个整数运算数包括随机数 r、保密密钥 s 和至少一个进一步的运算数(a, b)。通过把一公开密钥与保密密钥 s 相关联而选择了加密系统以后，加密单元配备了加密数据序列的发生器，其中所述公开密钥包括进行乘法运算的集合 G 的第一元素 g，每个加密数据序列都包括作为所述组件的运算数的随机数 r、以及取决于集合 G 的元素 g^r 的值 x，由单元与整数 y 一起给出。

组件可以由一个或多个电路部分或一个或多个软件模块所组成，该组件应用非常快速执行的基本加密项，这对于大量不同的加密方案都是共同有利的：验证、签名、密钥交换机制等等，它们使用了分集数学对象（可能定义多种离散对数函数的集合 G 和乘法运算）。

因为该组件对于大量机制是共同的，因此能更好地降低工业研发和制造成本。适用于该组件的基本单元（例如芯片卡）可以大量有利地生产，只要这些单元适用于有关族的所有方案，并且它们通常能实现这些或这种应用所需求的性能。

更具体地说，公开密钥还包括集合 G 的元素 v，使得 $v=g^s$ 或 $v=g^{-s}$。该方法使加密单元能根据广义的离散对数问题而应用整个方案族。这个问题广义上如下规定：令 G 是进行乘法运算（即根据两个元素 a 和 b，联系成表示为“a.b”的元素，或简单为“ab”的函数，称为 a 和 b 的乘积）的集合，g 是 G 的一个元素，u 是一个(大)整数，而 w 是 G 的元素，定义为 $w=g^u$(即根据 g 的出现 u 次的乘积 gg...g)；在实践中不可能从 g 和 w 中重现 u。

欧洲专利号 0666664 描述了这种类型的一个示例性电子签名方案，其中 G 是

大于等于0且小于n的一组整数，乘法元素是整数的普通乘积，模n。

根据本发明的方法，如果它发生，则对于给定的集合G和某一乘法运算而言，发现了比已知算法更为有效的离散对数计算算法，它足以改变其中执行计算的集合以及/或者乘法运算，以便重现期望的安全性级别。

离散对数问题可以在进行运算的任何集合内先验地规定。然而，为了使它能在短时间内执行指数计算并且提供小尺寸的结果，需要某些属性，譬如目前最适合的集合是组。除了其它属性以外，组总是包含一个中性元素，即表示为ε（或简单为1）的元素，使得乘积ε.a和a.ε都等于a，对于任何元素a都一样。此外，每个元素在组中有一个倒数，表示为a^{-1}，即使乘积$a^{-1}.a$和$a.a^{-1}$都等于ε的元素。加密中所使用的组的一般示例为整数和椭圆曲线的圈或范围。

因此可能定义一个加密组件，该组件不以任何方式依赖于有关的组，或者依赖于考虑中的集合G。这首先意味着这个组件不作用于集合自身的元素。这还意味着它不依赖于组的特性，也不依赖于考虑中的元素g，尤其是在G内g的数量级上，即满足$g^q=\varepsilon$的最小非零整数q（如果存在）。

在本发明的优选实施例中，组件所作用的组合仅由少量的整数间加法、减法和乘法所组成，没有一个与G和g的特性有任何联系。特别是，这个组合的形式可以是y=ar+bs的形式，其中a和b是两个进一步的整数运算数。进一步的简化使得a=1或b=1。

选择这个组件的一个优点是它的速度：如果仅仅要执行少量乘法（1或2），则组件会有高速度（几微秒），并且可以被结合在任何环境中，尤其结合在低成本的微处理器卡中。

通过把一随机数与一模数相关联来计算集合G上的指数，可以构造加密数据序列的发生器。

然而，在本方法的优选实施例中，加密数据序列的发生器包括一可编程存储器，用于接收预先计算的对$\{r,x\}$或$\{r,g^r\}$。这样，加密单元可以完全独立于集合G以及所采用的乘法元素而被完整地制造。仅仅需要把保密密钥s和预先计算的对$\{r,x\}$或$\{r,g^r\}$的某些数写入可编程存储器。在运算中，共同的组件会执行在加密单元级所需的唯一计算。

可以自主地使用该单元能进一步改进研发和制造成本的降低，这是由于在各种目标应用中可以使用相同的电路（而不仅仅是电路的相同部分）。此外，组件执行得很快能将其安装在非常低成本的电路中，因此在自主模式中，被接触或不接触

地安装在非常廉价的单元中，比如常规的微处理器卡。

这种自主性的还有一个优点是也许能够改变加密方案，例如由于后者已经损坏（即由于已经发现大大降低它所提供的安全性级别的攻击），而无须研发并制造另一电路，从而节约了生产力。

此外，如果单元使用值 x，其长度不会随时间而改变（例如由于它从 g^r 的计算因此涉及预定义的哈希函数），则在保留相同的方案时，可能无须研发并制造另一电路而改变所使用的其他密钥的长度。

此外，在最后两种情况下，不仅没有理由研发并制造另一电路，但如果适当地设计了后者，则甚至无须改变包含它们的安全性装置（例如芯片卡），即使在采用了这些装置以后。由于在已经运转的安全性装置（或安全性装置自身）中改变电路或电路的程序总是非常昂贵的操作，因此这个优点是非常显著的。

本发明可被以下组织有利地使用：生产安全芯片的半导体制造商、从这些芯片制造安全性装置的工业，比如芯片嵌入（有接触或无接触的芯片卡）、以及采用这种设备的组织（银行、电信业、货车司机等），对这些组织而言，加密单元的替换会造成高研发、制造、管理或维护成本。

总之，本发明用离散对数问题产生了一族公开密钥加密方案，其中一个实体执行最多由少量整数加法、减法和乘法组成的计算，该计算对于该族的所有方案是共同的。该计算最好表示出要由该实体执行的大多数计算，由于大多数其他计算可以预先执行。

从参照附图的非限制性示例性实施例的下列描述中，本发明的其他特性和优点将变得更为明显，附图中图 1 和 4 是按照本发明生产的加密单元的示意图。

下面考虑实体验证协议族，扩展为消息验证和消息的数字签名，以及用于交换密钥的协议族，全部实现一共同的组件。假定由另一实体 B 所使用的实体 A 的公开密钥真实性已经在前面由该实体 B 所确认。

令 G 是进行乘法运算的集合，g 是 G 的一个元素。实体 A 的保密密钥是整数 s。应该注意到，这个整数 s 的大小（它的基 2 分解的比特数）独立于 G 和 g。与实体 A 的 s 相关联的公开密钥是对 $\{g, v\}$，其中 $v=g^s$。

在本发明的示例性实施例中，实体 B 对实体 A 的验证如下进行：

1. A 随机地选取一整数 r，计算 $x=g^r$，并把 x 发送至 B；

2. B 随机地选取两个整数运算数 a 和 b，并把它们发送至 A；

3. A 计算 $y=ar+bs$，并把 y 发送至 B；

4．B 校验 $g^y=x^av^b$。

这个基本协议的许多变体是可行的，由于它适用于消息验证和数字消息签名：

— a 或 b 可以预先固定在一非零值（例如 a=1），该情况中这个运算数无须被发送，且组合 y=ar+bs 现在仅涉及乘法；

— y=ar+bs 可以用 y=ar-bs 代替，校验公式为：$g^yv^b=x^a$；

— y=ar+bs 可以用 y=bs-ar 代替，校验公式为：$g^yx^a=v^b$；

— y=ar+bs 可以用 y=-ar-bs 代替，校验公式为：$g^yx^av^b=1$

— 如果 G 是一个组，则保密密钥 s 的符号可以被反转，即取 $v=g^{-s}=(g^s)^{-1}$，其中校验公式变为：$g^yv^b=x^a$；这个选择当然可以与任一上述变化相结合；

— 在校验公式形式为 $g^yv^b=x$ 的每种情况下，假定 a=1，$x=g^r$ 可以用 $x=f(g^r)$ 来代替，其中 f 是函数，例如等于（或包括）加密哈希函数；校验公式于是变为：$f(g^yv^b)=x$；

— 同样在校验公式形式为 $g^yv^b=x$ 的每种情况下，假定 a=1，如果 M 是要由 A 证实的消息，则 $x=g^r$ 可以用 $x=f(g^r,M)$ 来代替，其中 f 是函数，例如等于（或包括）加密哈希函数；校验公式于是变为：$f(g^yv^b,M)=x$；所获得的协议为消息验证协议；

— 同样在校验公式形式为 $g^yv^b=x$ 的每种情况下，假定 a=1，如果 M 是要由 A 证实的消息，则 $x=g^r$ 可以用 $x=f(g^r,M)$ 来代替，其中 f 是函数，例如等于（或包括）加密哈希函数，于是计算 b=h(x)，其中 h 是没有特殊加密属性的函数，例如恒等式；在这种情况下，步骤 2 不再涉及实体 A；校验公式变为：$f(g^yv^{h(x)},M)=x$；所获得的协议为数字消息签名协议（在 G 是小于 n 的非负整数集合且运算为乘法模 n 的特定情况下，再现欧洲专利号 0666664 中描述的获得电子签名方案）。

注意到在步骤 3 中，实体 A 仅需执行一次整数加法以及一次或两次整数乘法。还注意到该组合独立于所选的集合 G。最后注意到，可以预先执行 A 需要执行的其它计算（$x=g^r$ 或 $f(g^r)$）。因此可能预先计算某些数量的 g^r 值（应用或不应用函数 f），然后把它们与相应的随机数 r 一起存储在可编程存储器中。

根据相同的参数，补充实体 B 的私有密钥 s'和相关的公开密钥 g'，v'，按照与实体 A 相同的规则获得，$g'=g$：$v'=g^{s'}$，密钥交换协议可以定义如下：

1．A 随机地选取整数 r，计算 $x=g^r$ 并把 x 发送至 B；A 计算共有密钥 $K=v'^r(=g^{s'r})$；

2．B 随机地选取两个整数运算数 a 和 b 并把它们发送至 A；

3．A 计算 y=ar+bs 并把 y 发送至 B。

4．B 校验 $g^y=x^av^b$。B 计算共有密钥：$K=x^{s'}(=g^{rs'})$

该协议一方面能按照 Diffie-Hellman 方案交换密钥，另一方面密钥交换在任一侧被验证。共有密钥 K 也可以被计算为 v'^{r} 的预定函数。

又注意到在步骤 3 中，实体 A 仅需执行一次整数加法以及一次或两次整数乘法。还注意到，该组合独立于所选的集合 G。最后注意到，必须预先执行 A 需执行的其他计算。因此可能预先计算某些数量的 x 值和 K 值，然后把它们存储在可编程存储器中。

因此，通过开发实现单独函数 y=ar+bs（或者上述替代之一）的程序或电路，获得了基本的软件或硬件程序块，它们可用于不同的加密方案中，实现诸如验证、密钥交换等不同的任务。实现给定任务的方案甚至可以在安全性设备的寿命期间被修改，所述安全性装置包括该程序或者该电路。例如，可能用另一个方案来代替该验证方案，或者保持相同的方案但改变其中执行计算的集合或组 G。实际上，这些修改仅会影响预先计算的值，而不影响组件自身。

图 1 图解地示出按照本发明生产的示例性加密单元 A。该单元由具有区域 10 的芯片所组成，到区域 10 的访问受到本领域技术人员公知的技术所保护。

被保护的区域 10 包括可编程存储器 11，用于一方面接收单元 A 的保密密钥 s（区域 12），另一方面一旦定义了集合 G 及其乘法运算（区域 13），接收独立于 s 确定的对 $\{r, g^r\}$。被保护的区域 10 还包括组件 15，用于计算整数 y=ar+bs，作为以下几者的函数：从存储器区域 13 接收的随机数 r、从存储器区域 12 接收的保密密钥 s 以及由控制模块 16 提供的两个进一步运算数 a、b。

在区域 13 内存储几个对 $\{r, g^r\}$ 的各种方式是可行的。每个 r 值和每个 g^r 值都可以被全部存储在一表格中，该表在相同对的 r 值和 g^r 值之间相关匹配。在具有受限存储器尺寸的微型电路中有利的是，一个简单的索引与每个 g^r 值相关联，以便节约存储几个 r 值所需的存储器空间，该空间一般很大。各个 r 值是通过伪随机发生器从初始值 r_0 以及从相应的索引而被预先计算的，以便预先计算并存储该索引的 g^r 值。于是，可编程存储器 11 包括伪随机发生器和最初的初始值 r_0，以便通过激活伪随机发生器而无须全部存储每个 r 值而从相应的索引接收每个 r 值，以便通过索引的优点而使它与 g^r 值相关联。

响应于远程实体 B 所发布的验证请求，控制模块 16 命令存储器区域 13 给出被定址到组件 15 的整数 r、以及集合 G 的相关元素 g^r，后者构成被发送到实体 B 的值 x。此外由控制模块 16 向组件 15 给出从实体 B 接收到的进一步运算数 a、b，然后由控制模块 16 把组件所返回的整数 y 传递给实体 B。获悉公开密钥 g、v 的实

体 B 将能够通过校验公式 $g^y=x^a v^b$ 的帮助来验证 A。

在图 2 的变体中，单元 A 用于验证消息 M。被保护的区域 10 和控制模块 16 基本上与图 1 的示例相同，固定 a=1。被保护的区域 10 增补一哈希模块 18，该模块应用了预定的加密哈希函数 f。该函数 f 的变元是来自存储器区域 13 的元素 g^r 以及由控制模块 16 提供的要被验证的消息 M。结果 x 被定址到控制模块 16，控制模块 16 将其传递至实体 B。

哈希模块 18 也可以存在于按照图 1 的实施例中，而没有变元 M（或者有该变元的缺省值），以便产生一密钥值 x，该值具有独立于集合 G 而规定的大小。

因此可见，相同的电路适用于两种应用。

对于按照图 3 的单元是相同的，该单元用于对消息 M 的签名，即独立于可能检查该签名的实体。如果哈希模块 18 给出的结果 x 采用整数形式，则它可以被提供给组件 15 作为运算数 b。还可能预先对其应用函数 h，如前所述。

在按照图 4 的实施例中，存储器区域 13 还把保密会话密钥 K 与每个随机数 r 相关联，保密会话密钥 K 被确定为实体 B 的公开密钥 g、v'（因此必须预先知道）的函数：$K=v'^r$。这个会话密钥 K 被定址到一保密密钥加密单元 20，该单元 20 按照对称加密算法以常规方式工作，以便可用于与实体 B 通信。后者通过校验公式 $g^y=x^a v^b$ 或者前述变体之一的帮助确保了保密密钥 K 的完整性。

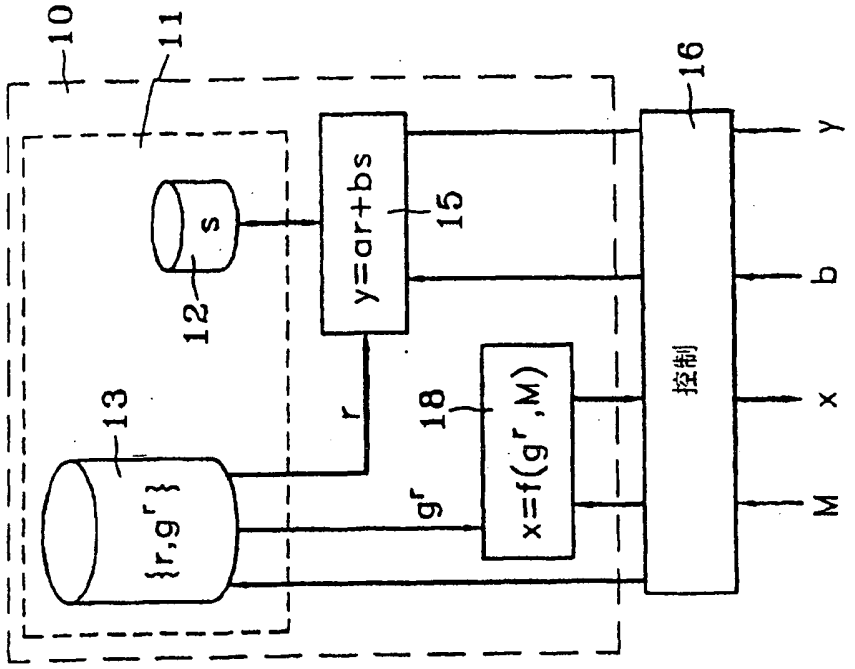

10
11
16
y
s
12
y=ar+bs
15
b
13
r
18
x
{r,g^r}
g^r
x=f(g^r,M)
控制
M

图 2

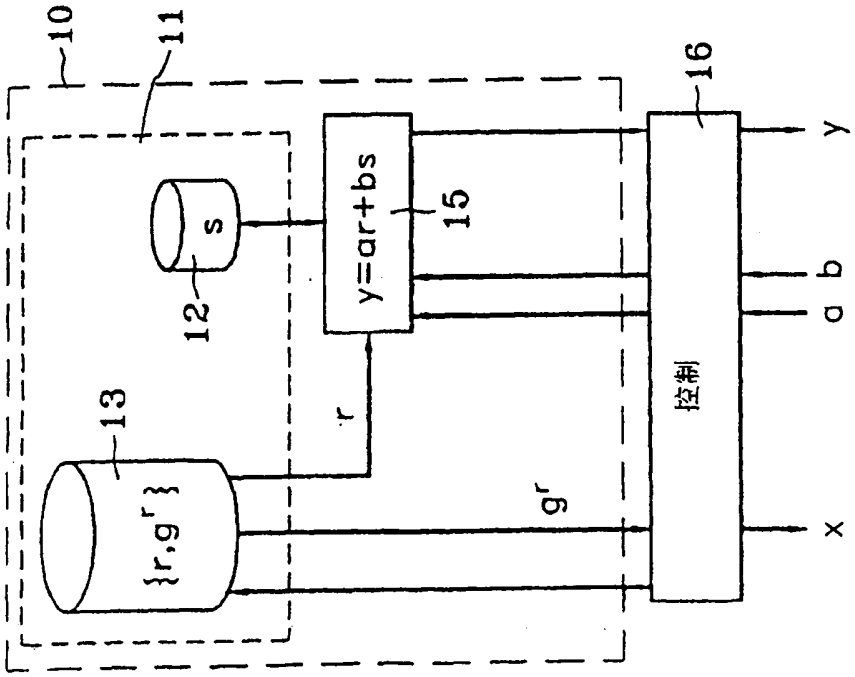

10
11
16
y
s
12
y=ar+bs
15
a b
13
r
{r,g^r}
g^r
控制
x

图 1

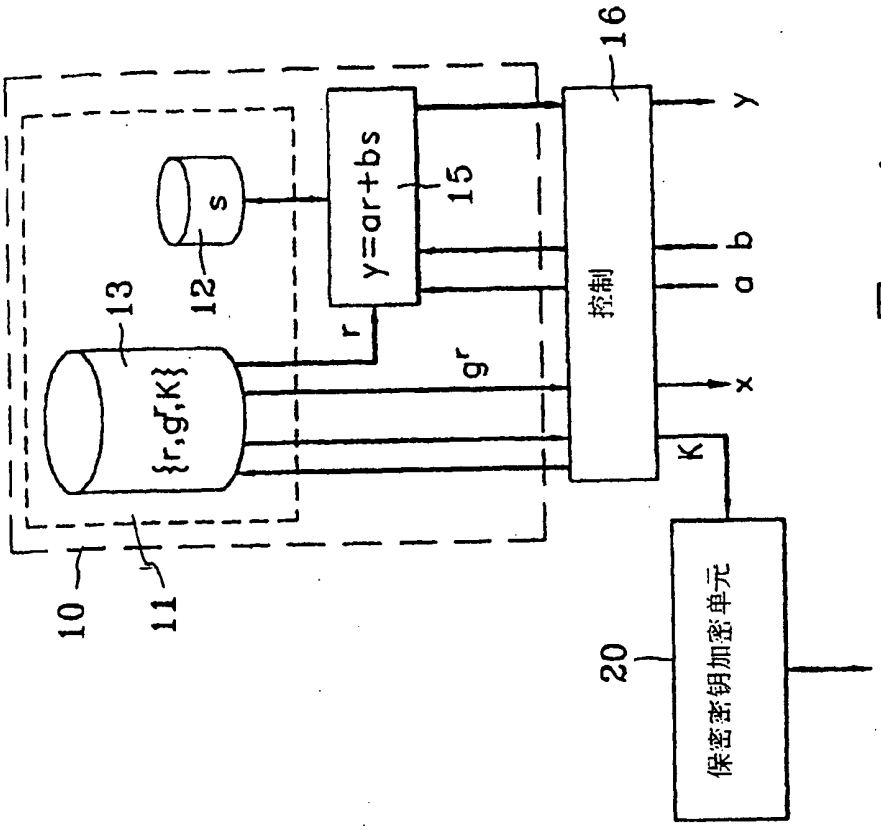

10
11
13
{r,g^r,K}
12
s
y=ar+bs
15
16
r
g^r
控制
y
a b
x
K
20
保密密钥加密单元

图 4

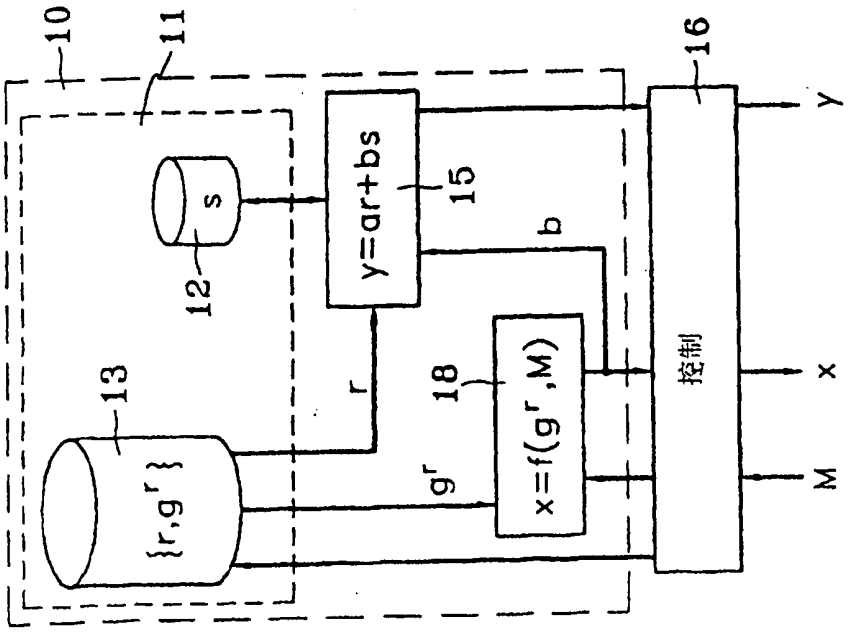

10
11
13
{r,g^r}
12
s
y=ar+bs
15
16
r
18
x=f(g^r,M)
g^r
b
控制
y
x
M

图 3