[19] 中华人民共和国国家知识产权局

[51] Int. Cl[7]
H04L 9/08

[12] 发明专利申请公开说明书

[21] 申请号 03805912.6

[43] 公开日 2005 年 7 月 20 日

[11] 公开号 CN 1643840A

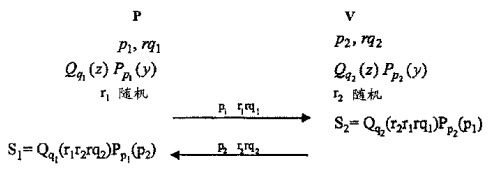
[22] 申请日 2003.2.14 [21] 申请号 03805912.6
[30] 优先权
[32] 2002. 3.13 [33] EP [31] 02075983.3
[86] 国际申请 PCT/IB2003/000655 2003.2.14
[87] 国际公布 WO2003/077470 英 2003.9.18
[85] 进入国家阶段日期 2004.9.13
[71] 申请人 皇家飞利浦电子股份有限公司
地址 荷兰艾恩德霍芬
[72] 发明人 P·T·图尔斯
T·A·M·科维纳亚
G·J·施里詹 M·E·范蒂克
[74] 专利代理机构 中国专利代理(香港)有限公司
代理人 程天正 刘 杰

权利要求书 3 页 说明书 19 页 附图 4 页

[54] 发明名称 基于多项式的多用户密钥生成验证方法和系统

[57] 摘要

一种在第一方和第二方之间产生一个共有秘密的方法，其中所述各方优选是依照数字权利管理(DRM)框架运作的家用网络(100)中的设备(101～105)。这些设备使用由受信任的第三方先前发布的参数以及从另一方获取的参数来对两个多项式 P(x，y)和 Q(x，y)的乘积进行求值，由此计算出所述共有秘密。优选地，各方随后使用零知识协议或是基于提交的协议来检验另一方是否产生了相同的秘密。所述方法尤其适合那些功率很低的设备，例如盘内芯片类设备。



ISSN1008-4274

1. 一种在第一方与第二方之间产生一个共有秘密的方法，其中第一方持有一个数值 p_1 和一个由数值 p_1 固定的第一自变量的对称多项式 P（x，y），并且第一方执行以下步骤：将数值 p_1 发送到第二方，从第二方那里接受数值 p_2，通过用 p_2 对多项式 P（p1，y）进行求值来计算共有秘密 S_1，其特征在于第一方额外持有一个数值 q_1 以及一个由数值 q_1 固定的第一自变量的对称多项式 Q（x，z），并且所述第一方还执行以下步骤：将 q_1 发送到第二方，从第二方那里接收一个数值 q_2，以及依照 $S_1=Q(q_1, q_2)\cdot P(p_1, p_2)$ 来计算秘密 S_1。

2. 权利要求 1 的方法，其中第一方还执行以下步骤：获取一个随机数 r_1，计算 $r_1\cdot q_1$，将 $r_1\cdot q_1$ 发送到第二方，从第二方那里接收 $r_2\cdot q_2$，以及依照 $S_1=Q(q_1, r_1\cdot r_2\cdot q_2)\cdot P(p_1, p_2)$ 来计算秘密 S_1。

3. 权利要求 2 的方法，其中第一方持有与一个任选数值 r 相乘的数值 q_1，并且还持有乘积 $Q(q_1, z)\,P(p_1, y)$ 而不是各单独的多项式 $P(p_1, y)$ 和 $Q(q_1, z)$，并且第一方还执行以下步骤：计算 $r_1\cdot r\cdot q_1$，将 $r_1\cdot r\cdot q_1$ 发送到第二方，从第二方那里接收 $r_2\cdot r\cdot q_2$，以及依照 $S1=Q(q_1, r_1\cdot r_2\cdot r\cdot q_2)\,P(p_1, p_2)$ 来计算秘密 S_1。

4. 权利要求 1 的方法，其中第二方持有一个数值 p_2、一个数值 q_2、由数值 p_2 固定的第一自变量的对称多项式 P（x，y）、由数值 q_2 固定的第一自变量的对称多项式 Q（x，z），并且第二方将会执行以下步骤：将 q_2 发送到第一方，从第一方那里接收 q_1，以及依照 $S_2=Q(q_2, q_1)\cdot P(p_2, p_1)$ 来计算秘密 S_2，由此，如果秘密 S_2 等于秘密 S_1，则生成了共有秘密。

5. 权利要求 1 的方法，其中受信任的第三方执行以下步骤：选择一个对称的（n+1）×（n+1）矩阵 T，通过使用来自矩阵 T 的各矩阵项作为多项式 P 的相应系数来构造多项式 P，构造多项式 Q（x，y），选择数值 p_1、数值 p_2、数值 q_1 和数值 q_2，将数值 p_1、数值 q_1、由数值 p_1 固定的第一自变量的多项式 P（x，y）以及由数值 q_1 固定的第一自变量的多项式 Q（x，z）发送到第一方，以及将数值 p_2、数值 q_2、由数值 p_2 固定的第一自变量的多项式 P（x，y）以及由数值 q_2 固定的第一自变量的多项式 Q（x，z）发送到第二方。

6. 权利要求 5 的方法，其中受信任的第三方还任选一个数值 r，

并且将替代数值 q_1 的数值 $r \cdot q_1$ 以及替代各单独的多项式 P（p_1，y）和 Q（q_1，z）的乘积 Q（q_1，z）P（p_1，y）发送到第一方，以及将替代数值 q_2 的数值 $r \cdot q_2$ 以及替代各单独的多项式 P（p_2，y）和 Q（q_2，z）的乘积 Q（q_2，z）P（p_2，y）发送到第二方。

7. 权利要求 5 的方法，其中受信任的第三方还执行以下步骤：

选择一个包含了 m 个数值 p_i 的集合，其中包含了数值 p_1 和 p_2，

根据那些从数值 p_i 的集合中构造的范得蒙矢量 $\vec{p}_i^{\,V}$ 的张量积 $\vec{p}_i^{\,V} \otimes \vec{p}_j^{\,V}$ 来计算空间 A，

从空间 A 的正交空间 $A^{\perp}$ 中选择矢量 $\vec{\gamma}_1$ 和矢量 $\vec{\gamma}_2$，从矢量 $\vec{\gamma}_1$ 中构造矩阵 $T_{\Gamma_1} = T + \Gamma_1$ 并且从矢量 $\vec{\gamma}_2$ 中构造矩阵 $T_{\Gamma_2} = T + \Gamma_2$，使用来自矩阵 T_{Γ_1} 的矩阵项构造一个多项式 $P^{\Gamma_1}(x,y)$，并且将由数值 p_1 固定的第一自变量的多项式 $P^{\Gamma_1}(x,y)$ 发送到第一方，以及

使用来自矩阵 T_{Γ_2} 的矩阵项构造一个多项式 $P^{\Gamma_2}(x,y)$，并且将由数值 p_2 固定的第一自变量的多项式 $P^{\Gamma_2}(x,y)$ 发送到第二方。

8. 权利要求 5 的方法，其中将数量为 m′ 的数值 p_i 分发给附加各方，其中 $m' < m$。

9. 权利要求 1 的方法，其中在将所生成的秘密 S1 和 S2 用作进一步通信之中的秘密密钥之前，第一方和第二方分别对其使用一个非线性函数。

10. 权利要求 9 的方法，其中将一个单向散列函数应用于所生成的秘密 S1 和 S2。

11. 权利要求 9 的方法，其中将多项式形式的非线性函数应用于所产生的秘密 S1 和 S2。

12. 权利要求 1 的方法，还包括检验第二方是否知道秘密 S1 的步骤。

13. 权利要求 12 的方法，其中第一方随后应用一个零知识协议来检验第二方是否知道秘密 S1。

14. 权利要求 12 的方法，其中第一方随后应用一个基于提交的协议来检验第二方是否知道秘密 S1。

15. 权利要求 14 的方法，其中第二方使用一个对称密码来加密一个随机挑战口令，并且将这个经过加密的随机挑战口令发送到第一方，第一方随后使用与提交函数相同的对称密码以保证对经过加密的

随机挑战口令进行解密。

16. 一种系统（100），其中包含了第一方（P）、第二方（V）以及受信任的第三方（TTP），所述系统被安排成执行上述任何一条权利要求的方法。

17. 一种设备（P），该设备被安排成在权利要求 16 的系统中充当第一方和/或第二方。

18. 权利要求 17 的设备，其中包含存储装置（303），用于以多项式 P 和多项式 Q 的相应系数的形式来储存多项式 P 和多项式 Q。

19. 一种计算机程序产品，用于使一个或多个处理器执行上述权利要求 1～15 中任何一条权利要求的方法。

基于多项式的多用户密钥生成验证方法和系统

本发明涉及一种在第一方与第二方之间产生共有秘密的方法，其中第一方持有一个数值 p_1 以及由数值 p_1 固定的第一自变量的对称多项式 P（x，y），并且第一方执行以下步骤：将数值 p_1 发送到第二方，从第二方那里接收数值 p_2，以及用 p_2 求值多项式 P（p_1，y）来计算秘密 S_1。

本发明还涉及一种被安排来执行这种方法的系统，其中所述系统包含了第一方、第二方和受信任的第三方，此外本发明还涉及被安排成起到系统中的第一或第二方的作用的设备，以及涉及一种计算机程序产品。

从 R. Blom 发表于 Advances in Cryptology-Proceedings of Crypto 82，231-236，1983 的文章“非公共密钥分发（Non-public key distribution）”已知依照前文的方法的一个实施例。

在数字通信网络和内容保护系统中，验证扮演了非常重要的角色。相互通信的设备需要确信彼此的可信赖性。它们不应将机密信息提供给不受信任的一方。而验证过程则通常以公共密钥技术为基础，这类技术需要大量的处理能力。然而在很多应用中却不能得到这种（处理）能力，在这种情况下则不能直接应用这些公共密钥技术。

有时也会提出一种以使用功耗小得多的对称密码为基础的解决方案。然而这些对称密码会遭遇到这样的缺陷，那就是这些对称密码需要在每个设备中都具有一个全局系统秘密，这对那些大量生产的产品来说并不理想。

在 CE 应用中，数字通信网络同样也在变得越来越普遍，并且驱动了对于廉价和低功率验证协议的需要。尽管这种功率约束对便携式 CE 设备和智能卡等而言一般都是成立的，但是如在由本申请的同一申请人所提交的国际专利申请 WO02/017316（代理人文档号 PHNL010233）中所述，这种功率约束在“盘内芯片”（CID）类产品中却是特别严格的。

支持 CID 的基本方法是在 CD 或 DVD 之类的载体上放置芯片，随后则将其使用于保护内容的目的。一旦该芯片确定播放器可信，那么

它会允许播放器播放内容（准许播放器访问它所承载的解扰密钥）。另一方面，播放器也不会播放不可信盘片上的任何内容。因此，播放器和 CID 全都需要某种用于进行验证的装置。

非常重要的是，应该指出，所述芯片在处理过程中只有很有限的功率（大约 0.5mW）并且由此无法执行很复杂的计算。这意味着不能直接使用公共密钥技术（例如 RSA 或 ElGamal）。CID 验证问题是 CE 领域中的验证问题的一个典型实例。

上文引用的 Blom 的文章公开了一种共有密钥或会议密钥生成方法，该方法使用的是一个以具有两个变量的对称多项式为基础的秘密共享协议。在图 1 中描述了这个协议。从根本上讲，名为证明者（prover）的一方（简写为 P）尝试让系统中名为检验者（verifier）的另一方（简写为 V）确信他知道一个同样为检验者所知的秘密。如果检验者确信，则验证了所述证明者。

在该系统中，受信任的第三方（TTP）选择一个对称的（n+1）×（n+1）矩阵 T，该矩阵的矩阵项 t_{ij} 表示的是具有两个变量的 n 次多项式 P 的相应系数，其中所述多项式是如下定义的：

$$P(x,y)=\sum_{i,j=0}^{n}t_{ij}x^{i}y^{j}$$

很明显，对多项式域中的所有 x 和 y 来说，P（x，y）=P（y，x）。通过将自变量 x 固定成的某个值（比方说 p）则可以将多项式 P 投影到具有一个自变量的 n 次多项式空间，即 P_p（y）=P（p，y）。从多项式 P 的定义、矩阵 T 的对称性和由此所得的 P（x，y）的对称性可以得出，对于所有的 p 和 q 来说，P_p（q）=P_q（p）。

根据 Blom 的文章，每一个需要能产生与另一个设备共有的秘密的设备都会接收一对（P_p（y），p），也就是固定于 p 的多项式 P 和用于从 P（x，y）产生 P_p（y）的数值 p。设备（P_p，p）和（P_q，q）之间的共享秘密是由 P_p（q）=P_q（p）给出的，其中所述等式是如下产生的：交换 p 和 q，并且对各多项式求值来为 P 产生 S_1 以及为 V 产生 S_2。

在这个方法中，全局秘密由矩阵 T 构成，由于该矩阵是对称的，

因此它具有 1/2（n+1）（n+2）个独立矩阵项。所述秘密的一部分是以一个相应数值 p 和多项式 P_p（y）的形式提供给每一方的，其中所述多项式具有 n+1 个采用如下形式的系数：

$$g_j = \sum_{i=0}^{n} t_{ij} p^i$$

这样一来，为各方提供了具有 1/2（n+1）（n+2）个未知数 t_{ij} 的 n+1 个线性等式，由此可以清楚了解到某一方不能获取全局秘密 T。只有在 n+1 方都以一个不同数值 p 来进行协作的情况下，才有可能获取矩阵 T。

这提出了已知协议的一个重大缺陷：如果有足够数量的多方协作，则就能获取全局秘密 T，除非不同数值 pi 的数目小于 n+1。但这意味着不同部分的数目受限于阻止泄露全局系统秘密 T 的多项式次数。此外，当两方进行通信时，它们总是产生相同的共有秘密。

本发明的一个目的是提供一种依照前文的方法，其中该方法允许将数量更多的全局秘密的不同部分分发给各方，但却不必增加多项式 P 的次数。

这个目的是依照本发明而在一种方法中实现的，该方法的特征在于：第一方额外持有一个数值 q_1 和一个由数值 q_1 固定的第一个自变量的对称多项式 Q（x，y），此外所述方法还执行以下步骤：将 q_1 发送给第二方，接收来自第二方的数值 q_2，以及依照 $S_1=Q(q_1, q_2)\cdot P(p_1, p_2)$ 来计算秘密 S_1。

虽然 p_i 的数值数目仍然仅限于 n，但是现在可以向各方分发数量更大的不同部分。在整个系统中，与 Blom 所描述的系统中的情况不同，q_i 的数值数目不受多项式 P 的次数限制，它仅仅受限于 Q 的域中的可能元素 q_i 的数目。由此可以有足够数量的 q_i 能为每一方都提供唯一的全局秘密部分。

在一个实施例中，第一方还执行以下步骤：获取一个随机数 r_1，计算 $r_1 \cdot q_1$，将 $r_1 \cdot q_1$ 发送到第二方，从第二方接收 $r_2 \cdot q_2$ 并且依照 $S_1=Q(q_1, r_1 \cdot r_2 \cdot q_2)\cdot P(p_1, p_2)$ 来计算秘密 S_1。随机数 r_1 和 r_2 隐藏了 q_1 和 q_2 的值，这样一来，使得窃听者或不兼容的（non-compliant）设备很难了解到 q_1 和 q_2 的某些情况。其次，r_1 和 r_2 的值导至多项式 P

和 Q 的求值结果倍增，由此，计算所得的秘密 S_1 和 S_2 也具有随机特性。这意味着如果稍后将 S_1 和 S_2 用作对称密码中的密钥，那么窃听者将是很难破译这种加密的。此外，现在在两个设备之间的每一个新会话上都可以生成一个不同的共有秘密。

在另一个实施例中，第一方持有与一个任选值 r 相乘的值 q_1 以及乘积 $Q(q_1, z)P(p_1, y)$，而不是各单独多项式 $P(p_1, y)$ 和 $Q(q_1, z)$，此外第一方还执行以下步骤：计算 $r_1 \cdot r \cdot q_1$，将 $r_1 \cdot r \cdot q_1$ 发送到第二方，从第二方那里接收 $r_2 \cdot r \cdot q_2$ 以及依照 $S1=Q(q_1, r_1 \cdot r_2 \cdot r \cdot q_2) \cdot P(p_1, p_2)$ 来计算秘密 S_1。这样一来，对可以使用设备并且试图了解全局秘密 T 和/或数值 q_1 或 q_2 的对手（adverary）来说，数值 q_1 和 q_2 都是隐藏的。

在另一个实施例中，在将一个非线性函数用作进一步通信之中的秘密密钥之前，第一方和第二方分别对所生成的秘密 S1 和 S2 使用了该非线性函数。优选地，这个非线性函数是作为一个单向散列函数来实现的，但是也可以采用多项式的形式。通过使用非线性函数，可以使得所述方案前向及反向安全。换句话说，即使攻击者设法得到了一个密钥，他也不能从所获取的密钥中导出先前或后续的密钥。

优选地，第一方随后检验第二方是否知道秘密 S_1。第一方可以通过应用一个零知识协议来检验第二方知道所述秘密 S_1。优选地，这个协议是带有公共值 e 和 m 的 Guillou-Quisquater 协议。这样一来则有如下优点，其中在本发明中，由于 Guillou-Quisquater 协议不允许对手预期一个挑战口令（challenge），因此对 e 的较低值来说，这种协议是非常安全的。此外，所述协议在通信和存储器使用方面也是非常有效的。

或者，第一方可以应用一个基于提交（commitment-based）的协议来检验第二方是否知道秘密 S_1。在执行所述方法的设备中，通过使用一个基于 DES、Lombok 或 AES 之类的对称密码的提交协议在功耗方面是非常有效的。优选地，第一方随后使用相同的对称密码作为提交函数而致力于对经过加密的随机挑战口令进行解密。由于可以将用于加密挑战口令的硬件和/或软件重用于执行提交函数，因此这样做具有降低实施方式复杂度的附加优点。

在从属权利要求中对其他有益的实施例进行了阐述。

通过参考附图中所显示的实施例，本发明的这些和其他方面将会变得显而易见并且可以得到清楚的说明，其中：

图 1 描述的是一个依照 Blom 的以具有两个变量的对称多项式为基础的秘密共享协议；

图 2 示意性显示了一个系统，其中包含了经由网络互联并被安排成依照本发明来进行操作的设备；

图 3 示意性显示了图 2 系统的概况，其中包含一个证明者、一个检验者和一个受信任的第三方；

图 4 描述的是证明者与检验者之间的秘密共享协议，其中所述协议是以各具有两个变量的两个对称多项式为基础的；

图 5 描述的是图 4 协议的一个变体，其中的两个多项式只在有限数量的点中是对称的；

图 6 描述的是 Guillou-Quisquater 协议；以及

图 7 描述的是一个基于提交的协议。

在所有这些附图中，相同的参考数字表示的是相似或相应的特征。图中显示的某些特征通常是以软件方式实现的，这样，这些特征代表了软件模块或对象之类的软件实体。

图 2 示意性显示了一个系统 100，其中包含了经由网络 110 互连的设备 101～105。在这个实施例中，系统 100 是一个家用网络。典型的数字家用网络包含多个设备，例如收音机、调谐器/解码器、CD 播放器、扬声器对、电视机、VCR、磁带录放音座（tape deck）等等。这些设备通常是相互连接的，由此允许诸如电视之类的其中一个设备控制诸如 VCR 之类的另一个设备。而一个例如调谐器/解码器或机顶盒（STB）之类的设备则通常是提供对其他设备的中央控制的中央设备。

通常包含音乐、歌曲、电影、电视节目、图片等的内容，是经由一个住宅网关或机顶盒 101 接收的。信源可以是连至宽带有线网络的连接、因特网连接、卫星下行链路等等。然后可以经由网络 110 而将内容传送到一个信宿，以便进行再现。举例来说，所述信宿可以是电视显示器 102、便携式显示设备 103、移动电话 104 和/或音频重放设备 105。

再现内容项的确切方法取决于设备类型和内容类型。例如在收音

机中，所述再现包括产生音频信号并且将其馈送到扬声器。对电视接收机来说，所述再现通常包括产生音频和视频信号并且将其馈送到显示屏幕和扬声器。对其他类型的内容来说，同样必须采取相似的恰当动作。此外，所述再现还可以包括解密或解扰接收信号、同步音频和视频信号等操作。

机顶盒 101 或是系统 100 中的其他任何设备都可以包括存储介质 S1（例如大小适当的硬盘），以允许记录以及随后重放所接收到的内容。存储器 S1 可以是与机顶盒 101 相连的某种个人数字记录器（PDR），例如 DVD+RW 记录器。此外还可以将保存在载体 120 中的内容提供给系统 100，其中所述载体可以是光盘（CD）或数字通用光盘（DVD）。

便携式显示设备 103 和移动电话 104 是通过使用基站 111 而无线连接到网络 110 的，其中举例来说，可以使用蓝牙或是 IEEE 802.11b。其他设备则是通过使用常规的有线连接相连接的。为了允许设备 101～105 进行交互，可以使用若干种互操作标准，这些标准允许不同的设备交换消息和信息以及相互进行控制。其中一种众所周知的标准是家庭音频/视频互操作性（HAVi）标准，其第 1.0 版发布于 2000 年 1 月并且可以在因特网上通过地址 http://www.havi.org/得到该标准。其他众所周知的标准是家用数字总线（D2B）标准、在 IEC1030 中描述的通信协议以及通用即插即用（http://www.upnp.org）。

通常，确保家庭网络中的设备 101～105 不产生内容的非授权拷贝是非常重要的。为此目的，有必要使用通常称为数字权利管理（DRM）系统的安全框架。

在一个这种框架中，从概念上讲，家庭网络是在一个有条件访问（CA）域和一个拷贝保护（CP）域中划分的。通常，信宿位于 CP 域中。这样一来，由于 CP 域中具有拷贝保护方案，因此可以确保在向接收器提供内容的时候不会产生所述内容的非授权拷贝。CP 域中的设备可以包括一个产生临时拷贝的存储介质，但是这类拷贝是不能从 CP 域中导出的。在本申请的同一申请人的国际专利申请 PCT/IB02/04803（代理人文档号 PHNL010880）中对这个框架进行了描述。

无论选择何种具体方法，在家用网络中实现所述安全框架的所有设备都会根据实现要求来实现所述安全框架。通过使用这个框架，这

些设备可以相互进行验证并且安全地分发内容。而对内容所进行的访问则是由安全系统来进行管理的。由此可以避免将不受保护的内容泄漏给非授权设备，并且阻止那些来自不受信任的设备的数据进入系统。

设备只将内容分发给其他那些它们先前已经成功验证了的设备是非常重要的。由此可以确保对手无法使用恶意设备来产生非授权拷贝。举例来说，由于只有经过授权的制造商才知道成功验证所需要的特定秘密，或者它们的设备具有一个由受信任的第三方所发布的证书，那么如果设备是由经过授权的制造商制造的，则该设备只能成功验证其自身。

秘密共享

在任何验证方案中都必须存在某种全局秘密或共有信息，并且想要向另一方验证自身的任何一方都必须至少与所述另一方共有某些信息。尽管从理论上讲是可以向每一个设备提供全局秘密的，但在实践中却并不建议采用这种做法：如果全局秘密成为已知（例如通过窃用一个设备），那么对手可以取代受信任的第三方（TTP）的角色，其中首先是所述第三方将全局秘密分发到受信任的各方的。这样一来，不兼容的设备将会进入系统并且初始系统的安全性将会受到损害，由此导致验证无用。此外由于整个全局秘密已知，因此将会无法检测到不兼容的设备。

一种解决这个问题的可能方法是秘密共享：受信任的每一方都会得到一部分全局秘密。这部分秘密足以使之向另一方验证其自身，但是会需要数量很大的诸部分来重建全局秘密（如果可能的话）。当一个设备受到损害的时候，只有一部分全局秘密会变成已知并且可以通过采取措施来废除这个设备。

本发明使用了一个秘密共享协议以使各方能够确定一个共有秘密。通常，各方随后会检验另一方是否知道秘密，参见以下章节“秘密检验”。然而，各方也有可能会在没有进行明确检查的情况下继续进行操作。例如，在这里可以将所述秘密用作一个加密密钥，以便对发送到另一方的某些信息进行加密。如果另一方不具有相同的秘密，那么他是不能解密信息的。由此隐性地授权了另一方。

图 3 示意性显示了图 2 系统的概况，其中包含了一个证明者 P、

一个检验者 V 以及一个受信任的第三方 TTP。依照本发明，检验者 V 希望通过使用从受信任的第三方 TTP 接收的信息来对证明者 P 进行验证。优选地，所述验证是相互的，由此证明者 P 也了解到检验者 V 也是可信的。

假设预先已将那些向证明者 P 验证检验者 V 所必需的信息从 TTP 分发到了各方 P 和 V。这可以在各方 P、V 和 TTP 之间的通信信道上进行。由此使得协议是动态的，并且允许在对手设法达成对先前分发的秘密的未授权访问的情况下很容易地更新信息。

证明者 P 和检验者 V 可以是具备了提供必要功能的芯片的设备(例如载体 120)，也可以是音频重放设备 105。在这种情况下，从 TTP 到证明者和检验者很可能没有通信信道。那么则必须预先对秘密进行分发，例如在制造载体 120 或设备 105 的工厂中。

证明者 P 包含了一个联网模块 301、一个密码处理器 302 和一个存储介质 303。通过使用联网模块 301，证明者 P 可以向检验者 V 发送数据，也可以从检验者 V 那里接收数据。联网模块 301 可以与网络 110 相连，也可以与检验者 V 建立直接连接（例如无线信道）。

密码处理器 302 被安排成执行依照本发明的方法。通常，这个处理器 302 是作为硬件与软件的组合来实现的，但是所述处理器也可以完全用硬件或软件方式来加以实现，例如作为软件模块或对象的集合。

举例来说，证明者 P 可以将多项式 P 和 Q 的系数保存在存储介质 303 中，但是也可将其用于保存某些希望在成功验证之后分发到检验者 V 的内容。存储介质 303 还可以用于保存那些从 TTP 接收的信息。为了增强系统安全性，与保存各单独多项式 P 和 Q 相反，在这里应改为保存乘积 $Q_q(z)P_p(y)$。

同样，检验者 V 包括一个联网模块 311、一个密码处理器 312 和一个功能与证明者 P 的存储器相对应的存储器 313。如果检验者 V 是作为带有盘内芯片的载体 120 来实现的，那么存储器 313 可以对应于任何（光）盘可用的存储器，但是优选将其保存在盘内芯片上的 ROM 中。

另外，证明者 P 和检验者 V 可以具有一个伪随机数发生器 304、314（硬件或软件形式），所述伪随机数发生器提供在加密方面很强的伪

随机数。这些数字都在依照本发明的方法的优选实施例中使用。现在通将会过参考图 4 和 5 来描述向检验者 V 验证证明者 P 的若干实施例进行描述。

使用两个对称多项式来产生一个共有秘密

图 4 描述的是依照本发明优选实施例的以分别具有两个变量的两个对称多项式为基础的秘密共享协议。在上文中已经参考图 1 而对部分设置以及由各方执行的步骤进行了描述，因此在此不再重复加以说明。

对称多项式 P 与一个对称多项式 Q（x，z）相乘，例如 Q（x，z）=x·z。除了以 p_1 固定多项式 P 之外，现在多项式 Q 同样是以 q_1 固定的。现在，证明者不再从 TTP 接收以 p_1 固定的多项式 P，而改以接收简化的多项式的乘积：

$$Q(q_1,z)P(p_1,y)=Q_{q_1}(z)P_{p_1}(y)$$

以及数值 p_1 和 q_1。同样，检验者接收的不是以 p_2 固定的多项式 P，而是简化多项式的乘积：

$$Q(q_2,z)P(p_2,y)=Q_{q_2}(Z)P_{p_2}(y)$$

以及数值 p_2 和 q_2。优选地，证明者和检验者以其系数的形式来保存所述多项式：

$$g_{1j}=q_1\sum_{i=0}^{n}t_{ij}p_1^i \text{和} g_{2j}=q_2\sum_{i=0}^{n}t_{ij}p_2^i$$

优选地，TTP 首先将 q_1 和 q_2 与一个随机因数 r 相乘。这样一来，对于可以未经授权而使用那些体现了证明者和/或检验者的设备的对手来说，数值 q1 和 q2 是隐藏的，由此防止其冒充经过授权的设备。

从上文中可以得到：

$$Q_{q_1}(rq_2)P_{p_1}(p_2)=q_1rq_2P(p_1,p_2)=q_2rq_1P(p_2,p_1)=Q_{q_2}(rq_1)P_{p_2}(p_1)$$

这表明：即使使用盲因数 r 来隐藏 q_i 的实际值，证明者与检验者也能够使用它们所具有的元素 p_i 和 q_i 以及从另一方那里接收的 p_i 和 q_i 来产生一个作为多项式 P 和 Q 的乘积的共有秘密。

如果我们现在将 p_i 的数值的数目限制成 n 或是更小，那么将不能获取多项式 P 和 Q 的系数。与 Blom 系统中的情况一样，在整个系统中，q_i 的值的数目不受多项式 P 的次数的限制，而只是受限于 Q 的域中可能元素 q_i 的数目。对足够数量的数值 q_i 来说，由此可以向每一方提供唯一的全局秘密部分。

如图 4 所示，在接收到多项式 P 和 Q 的乘积以及 p_i 和 q_i（或者 $r \cdot q_i$）的值之后，各方 P 和 V 现在尝试产生一个共有秘密。双方都交换其 p_i 和 q_i（或是 $r \cdot q_i$）的值并且计算各自的秘密 S_1 和 S_2。优选地，各方 P 和 V 首先产生各自的随机数 r_1 和 r_2。然后，它们将会分别计算 $r_1 \cdot q_1$ 和 $r_2 \cdot q_2$，并且对这些乘积而不是 q_1 和 q_2 自身的值进行交换。这样做有若干个优点，其中随机数 r_1 和 r_2 隐藏了 q_1 和 q_2 的值，这样使窃听者或不兼容的设备很难了解到关于 q_1 和 q_2 的信息。此外，这样做将会使任何一方（比方说证明者 P）都可以如下计算它的秘密 S_1：

$$S_1 = Q(q_1, r_1 \cdot r_2 \cdot q_2) \cdot P(p_1, p_2)$$

如果双方在将计算得到的 S1 和 S2 用作秘密密钥之前向所述 S1 和 S2 应用一个非线性函数，则可以实现本发明的更进一步的改进。优选地，非线性函数是作为一个单向散列函数实现的，但也可以采取多项式的形式。

使用受限的对称多项式来产生一个共有秘密

图 5 描述的是图 4 协议的一个变体，其中多项式 P 只在数量有限的点中是对称的。多项式 P 以一个对称矩阵 T 为基础，并且可以看到，在 P 的域中，对 x 和 y 的所有值来说，多项式 P（x，y）都是对称的。然而，如果使用了 n 个以上的不同数值 p_i，那么从理论上讲，对手是可以重建矩阵 T 的。因此，多项式 P 只需要在 m 个值 p_1，……，p_m 中保持对称，其中 $m \leq n$。为了说明如何构造那些只在数量有限的点中对称的多项式，我们首先给出一些定义。

两个 n 维矢量 $\vec{x} = (x_1, \ldots, x_n)$ 和 $\vec{y} = (y_1, \ldots, y_n)$ 的内积是由

$\langle \vec{x}, \vec{y} \rangle = \sum_{i=1}^{n} x_i y_i$ 给出的，$\vec{x}$ 和 $\vec{y}$ 的张量积 $\vec{x} \otimes \vec{y}$ 则是由 $\vec{x} \otimes \vec{y} = (x_1 \vec{y}, \ldots, x_n \vec{y})$ 给出的。

长度为 n+1 的范得蒙（Vandermonde）矢量 $\vec{p}^{V_n}$ 与由 $\vec{p}^{V_n} = (1, p, p^2, \ldots p^n)$ 给出的 p 相关联。除非另有说明，否则所有的范得蒙矢量的长度都是 n+1，并且为了便于标注，在这里我们将删除下标 n。在给出了具有 $m \leq n$ 个不同数值的子集 $\{p_1, \ldots, p_m\}$ 的情况下，我们构成范得蒙矢量 $\vec{p}_1^V, \ldots, \vec{p}_m^V$。这 m 个矢量是线性独立的。因此，这些矢量都是子空间 A 的基矢量。

接下来我们将对所有可能的张量积 $\vec{p}_i^V \otimes \vec{p}_j^V$ 进行考虑，其中 i，j=1，...，m。从张量张量中可以知道，这 m^2 个张量积构成了张量空间 $A = A \otimes A$ 的基。对所有矢量 $\vec{\gamma} \in A^{\perp}$ 来说，都有

$$\left\langle \vec{\gamma}, \vec{p}_i^V \otimes \vec{p}_j^V \right\rangle = 0$$

通过使用以上定义，可以将多项式 P（x，y）改写成一个内积：

$$P(x,y) = \left\langle \vec{t}, \vec{x}^V \otimes \vec{y}^V \right\rangle$$

其中 $\vec{t}$ 表示的是矢量（t_{00}，...，t_{0n}，t_{10}，...，t_{nn}）。也就是说，它包含了矩阵 T 的矩阵项。在其改写形式中，P 仍旧是对称的。

然后，我们选择 m 个不同元素 p_1，...，p_m。通过使用这些元素，我们构造范得蒙矢量 $\vec{p}_i^V$ 并且从这些范得蒙矢量中构造张量积 $\vec{p}_i^V \otimes \vec{p}_j^V$。随后如上所述，我们从空间 A 的正交空间 $A^{\perp}$ 中选择一个矢量 $\vec{\gamma}$。然后则可以在那些选自优选集合 $\{p_1, \ldots, p_m\}$ 的点中对多项式 P 的改写形式进行求值。可以将矢量 $\vec{\gamma}$ 添加给矢量 $\vec{t}$，并且由于 $\gamma \in A^{\perp}$，因此有：

$$P(p_i, p_j) = \left\langle \vec{t} + \vec{\gamma}, \vec{p}_i^V \otimes \vec{p}_j^V \right\rangle = \left\langle \vec{t}, \vec{p}_i^V \otimes \vec{p}_j^V \right\rangle + \left\langle \vec{\gamma}, \vec{p}_i^V \otimes \vec{p}_j^V \right\rangle = \left\langle \vec{t}, \vec{p}_i^V \otimes \vec{p}_j^V \right\rangle$$

换句话说，如果我们从矢量 $\vec{\gamma} = (\gamma_1, \ldots, \gamma_{(n+1)^2})$ 中导出一个矩阵

$$\Gamma=\begin{bmatrix}\gamma_1 & \gamma_{n+2} & \cdots & \gamma_{n^2+n+1}\\ \vdots & \vdots & \ddots & \vdots\\ \gamma_{n+1} & \gamma_{2n+2} & \cdots & \gamma_{(n+1)^2}\end{bmatrix}$$

并且将这个矩阵 Γ 添加给矩阵 T，那么对优选集合中的所有 p_i 和 p_j 来说，我们仍旧具有 P（p_i，p_j）=P（p_j，p_i）。

上述观测值由 TTP 使用，以便通过执行下列操作来设置所述系统：

1. TTP 选择一个随机对称的（n+1）×（n+1）矩阵 T 并且优选地选择一个任意值 r。

2. TTP 选择 m 个不同的随机元素 p_1，...，p_m，其中 m≤n。

3. TTP 从张量积 $\vec{p}_i^{\,V}\otimes\vec{p}_j^{\,V}$ 中计算空间 A。

4. TTP 从 m 个元素 p_1，...，p_m 中优选的选择前 m′<m 个元素。由此可以对系统进行更新（下文中将在“可更新性”这一节中对此加以说明）。

然后，TTP 可以发布某些设备，也就是将全局秘密的一部分提供给各设备，由此允许这些设备对具有一部分全局秘密的其他设备（相互）验证其自身。这种设备通常称为已鉴定设备或已授权设备。接下来，在相互验证了其他已鉴定设备之后，一个已鉴定设备通常会因为其与某个非授权设备所进行的验证失败而检测到一个非授权设备。

为了发布一个设备，TTP 将会执行以下步骤：

1. TTP 为设备 i 随机选择 $\vec{\gamma}_i\in A^{\perp}$ 并且从具有 m 个元素 p_1，...，p_m 的集合中随机选择 p_i，其中优选的是从具有 m′个元素的选定子集中选择 p_i。

2. TTP 从 $\vec{\gamma}_i$ 中产生一个矩阵 Γ_i 并且构成矩阵 $T_{\Gamma_i}=T+\Gamma_i$。

3. TTP 从 T_{Γ_i} 中构造二变量多项式 P（x，y）并且计算可以表示为 $T_{\Gamma_i}\vec{p}_i^{\,V}$ 单变量多项式 P（p_i，y）的系数。

4. TTP 将数值 p_i、$r\cdot q_i$ 以及矢量 $q_iT_{\Gamma_i}\vec{p}_i^{\,V}$ 分发给设备 i。

如图 5 所示，在接收了其各自的信息之后，各方 P 和 V 现在交换他们的数值 p_i 和 $r_i\cdot r\cdot q_i$，并且如下产生他们各自的秘密 S_1 和 S_2：

$$S_i=Q_{q_i}(r_ir_jrq_j)P_{p_i}^{\Gamma_i}(p_j)=r_ir_jrq_j\left\langle q_iT_{\Gamma_i}\vec{p}_i^{\,V},\vec{p}_j^{\,V}\right\rangle$$

如果 $S_1=S_2$，那么各方已经产生了一个共有秘密。在这里，各方既可以隐性推断出另一方也知道这个秘密，也可以显性检验另一方是否知道同一秘密。在下文中的“秘密检验”中对此进行了论述。

可更新性

对一个与系统 100 相似的系统而言，其任何一种验证或共有密钥生成方案中的重要方面都在于可更新性。TTP 可能希望定期更换设备 101～105 中安装的秘密，以便阻止那些已经能对原始秘密进行非授权访问的对手。

通过利用先前各章节中说明的特性，可以使用图 5 中描述的各实施例而将可更新性引入系统 100。一开始，TTP 仅仅使用元素 p_1，……，$p_{m'}$来发布设备，其中 $m'<m\leq n$，这样一来，$\vec{p}_i^V \otimes \vec{p}_j^V$ 将会跨越空间 A'，其中 $i,j\in\{1,\ldots,m'\}$。然而矩阵 $T_\Gamma = T + \Gamma$ 却使用了从 $\vec{\gamma}\in A^{\perp}$ 中导出的各 Γ。如果我们借助 $T_\Gamma \vec{p}_i^V$ 来表示设备 i 中保存的多项式，则所述设备将会包含一对 $(T_\Gamma \vec{p}_i^V, p_i)$。

现在我们假设对手能够以例如暴力开启设备之类的某种方式获取 m'个元素 p_i 以及某个设备多项式 $T_\Gamma \vec{p}_i^V$。对手现在可以产生一个新矢量 $\vec{\gamma}'\in A'^{\perp}$ 并且发布那些包含了 $((T_\Gamma+\Gamma')\vec{p}_i^V, p_i)$ 的设备。这些设备会与所有包含了数值 p_1，…，$p_{m'}$ 之一的兼容设备合作：对手的设备从一个兼容的设备中接收 $p_j\in\{p_1,\ldots,p_{m'}\}$ 并且求值

$$P(p_i,p_j)=\left\langle \vec{t}+\vec{\gamma}_i+\vec{\gamma}', \vec{p}_i^V\otimes\vec{p}_j^V\right\rangle=\left\langle \vec{t}, \vec{p}_i^V\otimes\vec{p}_j^V\right\rangle$$

并且第二方求值

$$P(p_j,p_i)=\left\langle \vec{t}+\vec{\gamma}_i, \vec{p}_j^V\otimes\vec{p}_i^V\right\rangle=\left\langle \vec{t}, \vec{p}_j^V\otimes\vec{p}_i^V\right\rangle=\left\langle \vec{t}, \vec{p}_i^V\otimes\vec{p}_j^V\right\rangle$$

这表明这两个求值是相同的。

如果 TTP 注意到这类设备是由一个对手发布的，那么 TTP 可以开始使用 $p_{m'+1}$，…，$p_{m''}$ 来发布设备，其中 $m'<m''\leq m$，由此 $\vec{p}_1^V\cdots\vec{p}_{m'}^V$ 的张量积将会跨越一个空间 A''。应该注意的是，$A''^{\perp}\subset A'^{\perp}$。因此，如果对手选择了 $\vec{\gamma}'\in A''^{\perp}$，那么这些新设备会与对手的设备合作。如果在 A'中随机地选择 $\vec{\gamma}'$，那么它处于 $A''^{\perp}$ 中的概率将会很小。

由此为系统提供了一定的可更新性：由 TTP 发布的新的兼容的设备与对手的设备合作的概率不高。可以更新系统的最大次数是 m-1<n，其中 n 是多项式 P 的次数。这种情况会在当每次更新添加 $p_i \in \{p_1,\ldots,p_m\}$ 中的一个数值的时候出现。

秘密检验

在各方各自独立产生秘密之后，所述协议的下一个步骤是检验另一方是否了解该秘密。如果其中一方可以向另一方证实他知道所述秘密，则所述此方对另一方是经验证的。此外，另一方同样可以向第一方验证其自身，从而实现相互验证。

在检验出证明者知道所述秘密之后，检验者然后可以使用秘密 S_1 向证明者安全地传送某些信息。例如，在这里可以使用 S_1 来加密那些访问加密内容所需要的加密密钥。而结果则可以被传送到证明者，证明者转而使用 S_2（如成功验证所证实的那样，所述 S_2 与 S_1 相同）来恢复加密密钥，然后则解密并访问所述加密内容。

目前存在若干种方法来检验某一方是否知道如上产生的秘密。其中有两个优选实施例是以零知识协议和基于提交的协议为基础的。

基于零知识的检验

首先将要论述的是基于零知识（ZK）协议的检验。在 1996 年由 CRC Press 出版的由 A. Menezes、P. Van Oorschot 和 S. Van Stone 编著的“应用密码学手册（Handbook of Applied Cryptography）”第 405～416 页中论述了 ZK 协议。由于 Guillou-Quisquater（GQ）零知识协议在存储器要求和通信方面非常有效，因此在一个优选实施例中使用了这个协议。此外也可以从本申请的同一受让人的 US 5，140，634（代理人文档号 PHQ87030）中可以了解到 GQ 协议。

如参考图 4 和 5 所述，P 和 V 双方都对其多项式进行了求值并且由此分别获得了数值 S1 和 S2。其中任何一方现在都必须以 ZK 方式向另一方证实他知道 Si。由于 GQ 协议是以公钥密码密码学为基础的，因此我们需要一个复合数 m=pq（这个复合数是两个素数 p 和 q 的乘积）与一个数字 e>1，以使得 gcd（e，（p-1）（q-1））=1。

P 将向 V 证实他知道 $S_2^e \bmod m$ 的 e 次方根。在图 6 中对 GQ 协议进行了描述，其中数值 e 和 m 是公开的。所述协议则依照下列步骤继续进行：

1. V 计算 $v=S_2^e$，

2. P 选择一个随机数 $r\in\{2,\ldots,m-1\}$ 并且将 r^e 发送到 V，

3. V 选择一个随机挑战口令 $c\in\{1,\ldots,e-1\}$ 并且将 c 发送到 P，

4. P 使用 $y=rS_1^c$ 来做出应答，

5. V 计算 y^e，当且仅当 $y^e=(rS_1^c)^e \bmod m=r^e v^c \bmod m=r^e(S_2^e)^c \bmod m=(rS_2^c)^e \bmod m$ 的时候，由于这意味着 $S_1=S_2$，因此 V 可以推断出 P 知道与 V 相同的秘密。

由于协议的 ZK 特性，因此 V 和窃听者都不知道关于 P 的秘密 S 的任何信息。一旦 V 接受 P，那么 P 和 V 的角色将会互换，并且 V 会向 P 显示他知道 S_1^e 的 e 次方根与 m 的模。由此 P 和 V 进行了相互验证。

协议的设置与文献中找到的内容略有不同：通常所发布的是 $v=S_2^e$，并且如果 P 预期一个挑战口令 c*，那么他可以发送 $z^e v^{-c^*}$ 作为第一消息并且在不了解 S_2 的情况下仍旧为 V 所接受。选择正确挑战口令的概率是 e^{-1}。在当前的设置中没有必要发布 $v=S_2^e$，这就使得 P 不能从所预期的挑战口令中计算出 v^{-c^*}，而这就将不当接受的概率降至 m^{-1}。因此，在这里可以将 e 选择为低至 2，从而有效地将 GQ 变换成 Fiat-Shamir 协议，只是在一个循环中有一个差错概率 m^{-1}。这意味着与例如 RSA 相比，设备只需要执行指数较小的模指数运算。

为了使之更为有效，可以设想一种使用了蒙哥马利表示的实施方式（参见 P. L. Montgomery 在 1985 年 4 月发表于 Mathematics of Computation 第 44 卷第 170 号第 519～521 页的“无需试除的模乘法（Modular multiplication without trial division）”）。

基于提交的检验

作为 ZK 协议的一种替换方案，可以使用基于提交的协议来允许一方检验另一方是否知道秘密。这种方法的优点是可以使用对称密钥密码加密，其中所述密码加密可以得到非常有效的实施。

与先前情况相反，P 和 V 双方同时起到检验者和证明者的作用，由此使得协议在通信方面非常有效。如前所述，P 计算 S_1 并且 V 计算 S_2。所述协议（参见图 7）则经历以下步骤：

1. V 选择一个长度与对称密码的块长度相匹配的随机数 r。

2. V 使用一个对称密码来加密 r（其中 S2 充当一个密钥），以及将编密码 E_{S2}（r）发送到 P，

3. P 使用 S1 来解密消息。结果是 $r' = D_{S_1}(E_{S_2}(r))$。

4. P 选择一个随机数 R 并且将一个关于 r'的提交发送到 V。如下所述，所述提交是作为一个函数 commit（R，r'）获取的。

5. V 将 r 发送到 P，P 则检查是否 r'=r，如果没有出现这种情况，则停止与 V 的进一步通信。

6. P 将 r'和 R 发送到 V。V 打开该提交并且检查是否 r'=r，如果检查不被满足，则停止与 P 的进一步通信。

commit 函数应该实现提交的绑定（binding）和隐藏特性。绑定指的是 P 改变提交中的值 r'的能力。对 P 来说，要想找到值 R'以使得 commit（R，r'）=commit（R'，r）是非常困难或不可能的。隐藏特性指的是 V 在接收到 commit（R，r'）之后获取关于 r'的信息的能力。在实践中通常将密码散列函数或单向函数用作 commit 函数。

在这个设置中也可以将用于加密 r 的对称密码用作 commit 函数。由于 V 不能在不了解随机选定的 R 的情况下得到关于 r'的信息，而与 V 的计算能力的大小无关，因此可以很平常地满足所述隐藏特性。因此，所述提交是无条件隐藏的。绑定特性则是从如下事实中得出的，其中对一个对称密码来说，目前已经知道 $E_K(x)=z$ 是一个 K 的单向函数（其中 x 和 z 都是已知的）：在给出了 $E_R(r')$ 和 r 的情况下，并不知道如何在少于 2^{55} 个操作中找到一个使得 $E_{R'}(r)=E_R(r')$ 的 R'值。因此，所述提交是通过计算绑定的。

接下来我们考虑的是协议的完整性和健全性。完整性指的是双方都正确执行协议以及 $S_1=S_2$ 的情况。由此借助检查以及对称密码的对称特性得出，当 $S_1=S_2$ 时，它们将会发现 r=r'。

健全性指的是在 P 不知道 S_1 或 V 不知道 S_2 时相互接受的情况。要想被不正当地接受，P 可以将任何值 z 作为一个提交发送到 V。在从 V 接收了 r 之后，P 必须找出一个值 R'，以使得 commit（R'，r）$=E_R(r)=z$。如上所述，$E_K(X)=z$ 在给出了 x 和 z 的情况下是 K 中的单向函数，这使得找到 R'成为一个非常困难的问题。

同样，如果 V 不知道 S_2，那么他可以针对 P 而选择任何一个值 z，其中 P 将会用 $E_R(D_{S2}(z))$ 来做出应答。为了被接受，V 必须获取 $D_{S2}(Z)$，但是因为随机值 R 而使得所述提交是无条件隐藏的，因此所述获取是非常困难的。如果 S_1 正好是一个脆弱的 DES 加密密钥，那么倘

若 V 通过选择 z 而使得 D_{S1}（z）=z，则 V 将被接受。对一个脆弱的密钥来说，存在 2^{32} 个这类固定点，而被 P 不正当地接受的概率是 $2^{32}/2^{64}=2^{-32}$。

本发明的某些优点

在那些执行依照本发明的方法的设备中，所述方法极大节约了必要的能量（功率），此外与基于 RSA 的验证相比，该方法极大地节省了处理时间。

通常，功耗取决于实施方式的架构。例如，通过改变架构，可以在时钟速度和功率消耗之间的折衷。而第二个重要因素则在于所使用的技术：与较早的技术相比，通常，最小形体尺寸较小且电源电压较低的现代技术所需要的功率更少。

下表根据 n（多项式次数）、k（一个值的比特长度）、1（GQ 模数的比特长度）和 h（RSA 模数的比特长度）而给出了对于协议不同部分的必要工作的估算。所估算的工作是按照单精度乘法（sp-mults），也就是在两个 k 比特数的乘法环境中进行的两个比特的乘法，来表述的。

子协议	需要的工作
多项式求值	k^2（n+3） sp-mult
GQ 协议	201^2 sp-mult
提交协议	100，000 门变换
RSA 协议	$3/4h^3$ sp-mult

下表显示的是在可用功率为 0.5mW 的盘内芯片应用中使用本发明的时候估算得到的对于 n、k、1 和 h 的 n 个值的以焦耳为单位的各子协议需要的能量以及处理时间量。

	n=128 k=64 l=512 h=512	n=512 k=64 l=512 h=512	n=128 k=128 l=1024 h=1024	n=512 k=128 l=1024 h=1024	n=2048 k=64 l=1024 h=1024
多项式	471n	1.86μ	1.86μ	7.35μ	7.47μ
GQ	4.51μ	4.51μ	18.0μ	18.0μ	18.0μ
提交	2n	2n	2n	2n	2n
多项式+GQ	4.98μ	6.37μ	19.9μ	25.4μ	25.5μ (52ms)
多项式+提交	473n	1.87μ	1.87μ	7.36μ	7.47μ (15 ms)
RSA	86.5μ	86.5μ	692μ	692μ	692μ (1.4s)

应该注意的是，上述值基于的是对每个 sp-mult 所需能量的估计。实际能量则取决于设计过程中的选定架构、布局、最优化目标（例如功率或速度）等等。然而，上表中的数据洞察了不同协议所需要的能量比率。在最后一列中可以看到，即使对那些次数为 2048 和 64 比特值的多项式来说，与 RSA 相比，新协议的效率要高出 30 到 100 倍。

在 CID 最多具有 0.5mW 的可用功率的特例中，我们导出 RSA 协议需要大约 1 秒的时间，而基于对称多项式的协议最多只需要 52ms 的时间。

应该注意的是，上述实施例是对本发明进行说明而不是对其加以限制，在不脱离所附权利要求书范围的情况下，本领域技术人员能够设计出多种替换实施例。虽然上文在内容保护和数字权利管理方面阐述了验证方法，但是毫无疑问，本发明并不局限于这一方面。

本发明可以视为是用于在任何一对组件和/或设备之间的接口上进行验证的通用构件块，在低功耗非常重要的时候更是如此。这样，举例来说，本发明还可以应用于 CD2、机顶盒、无线智能卡、有线或无线网络等等。当个人检验者需要使用两个相应的互连设备来验证一个个人证明者的时候也可以使用本发明。

很明显，在上文中使用了术语“随机数”或“任选数字”，这其中包含了在无论是否具有来自真正随机事件的种子值（seed value）

的情况下通过使用以硬件和/或软件方式实现的伪随机数发生器而选择的数字。本发明的安全性极大依赖于伪随机数发生器的质量。

在权利要求书中，任何置于括号中的参考符号不应视为是对权利要求进行限制。单词“包含”并不排除在权利要求所列举的元件或步骤之外还存在着其他元件或步骤。元件之前的单词“一个”并不排除存在多个此类元件。此外，本发明既可以借助于包含若干不同元件的硬件来实施，也可以借助于一个经过适当编程的计算机来实施。

在列举了若干种装置的设备权利要求中，其中有几种装置可以通过同一个硬件项来实现。在互不相同的从属权利要求中阐述了某些措施，然而这一事实并不表示不能有效使用这些措施的组合。

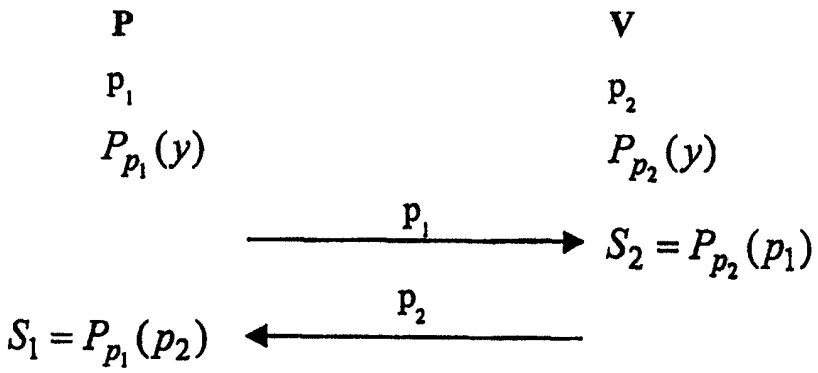

P
V
p_1
p_2
$P_{p_1}(y)$
$P_{p_2}(y)$
p_1
$S_2 = P_{p_2}(p_1)$
p_2
$S_1 = P_{p_1}(p_2)$

图 1

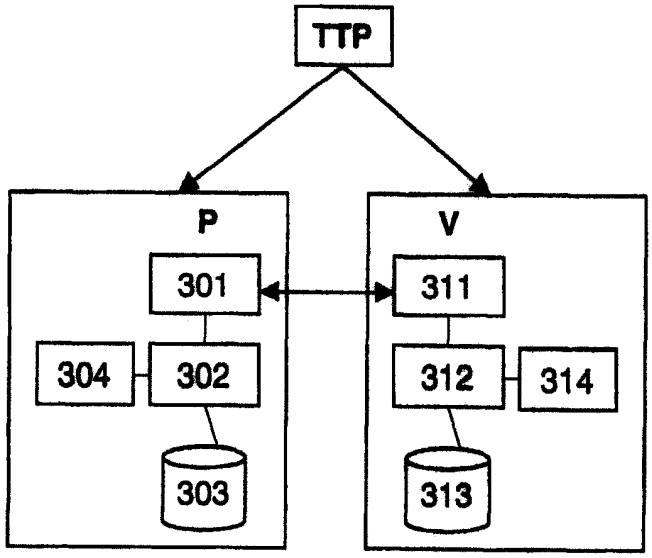

TTP
P
V
301
311
304
302
312
314
303
313

图 3

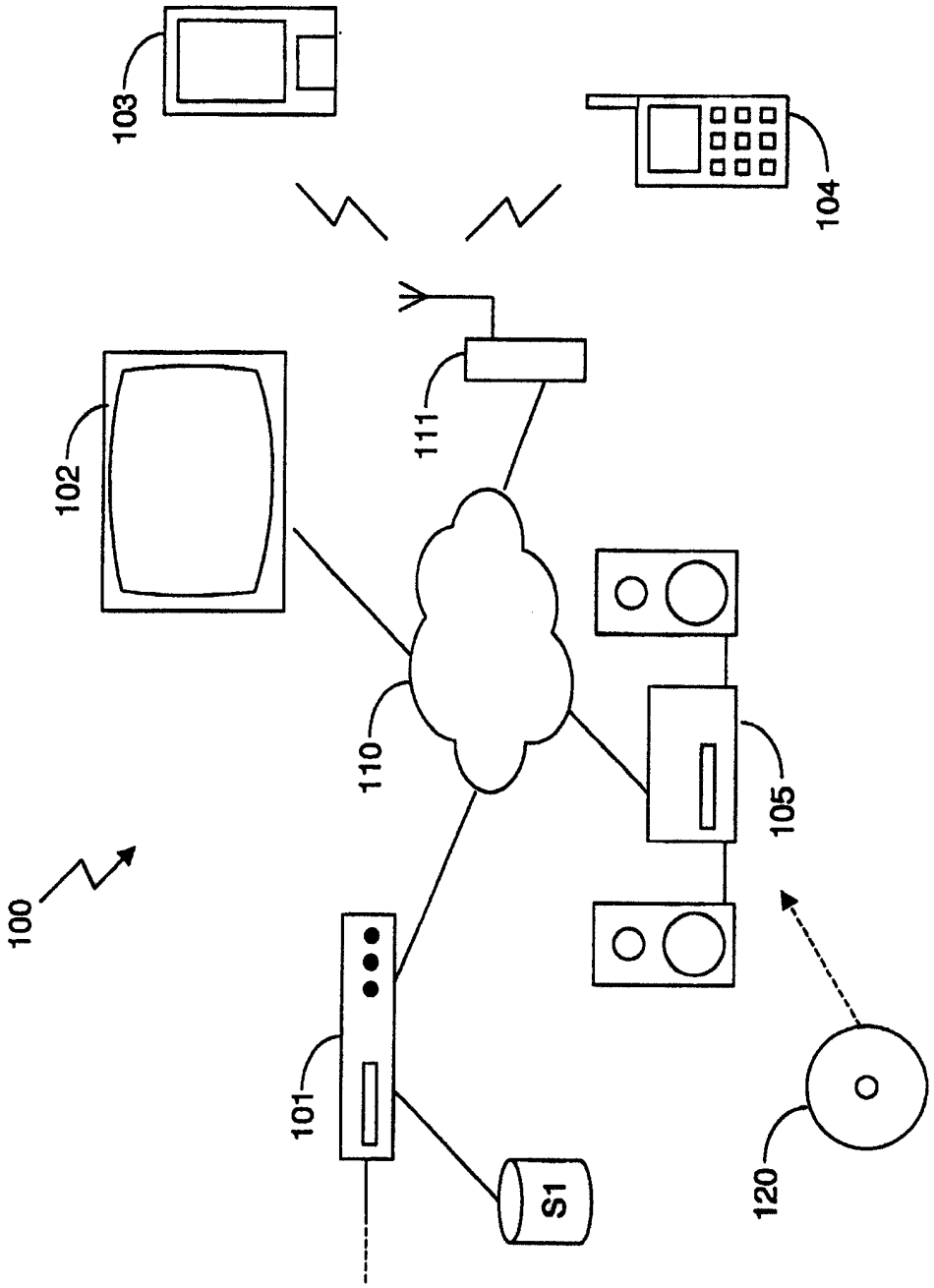


图 2

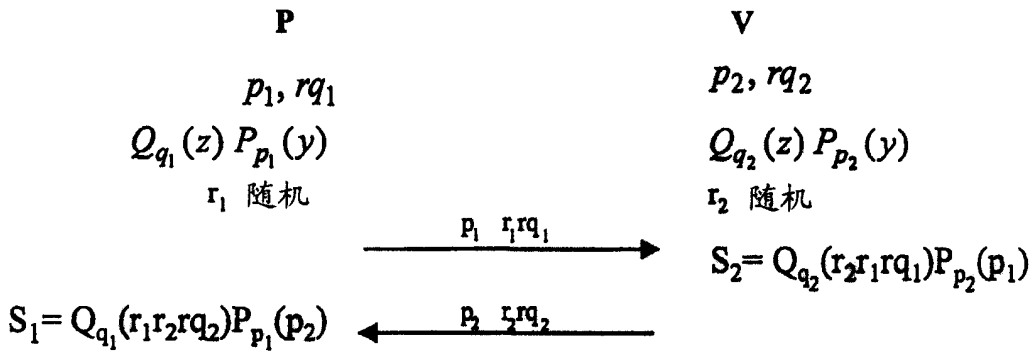

P
V
p_1, rq_1
$Q_{q_1}(z)\ P_{p_1}(y)$
r1 随机
p_2, rq_2
$Q_{q_2}(z)\ P_{p_2}(y)$
r2 随机
p1 r1rq1
S2= Qq2(r2r1rq1)Pp2(p1)
S1= Qq1(r1r2rq2)Pp1(p2)
p2 r2rq2

图 4

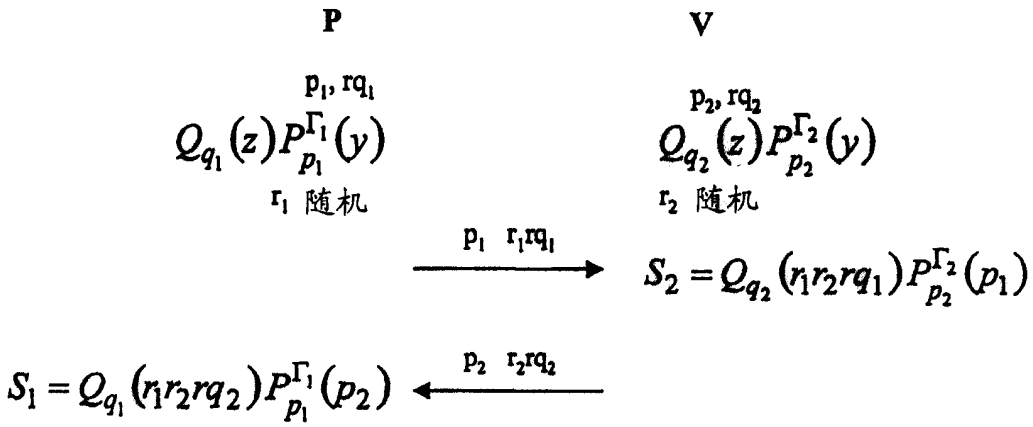

P
V
p_1, rq_1
$Q_{q_1}(z)P_{p_1}^{\Gamma_1}(y)$
r1 随机
p_2, rq_2
$Q_{q_2}(z)P_{p_2}^{\Gamma_2}(y)$
r2 随机
p1 r1rq1
$S_2 = Q_{q_2}(r_1r_2rq_1)P_{p_2}^{\Gamma_2}(p_1)$
$S_1 = Q_{q_1}(r_1r_2rq_2)P_{p_1}^{\Gamma_1}(p_2)$
p2 r2rq2

图 5

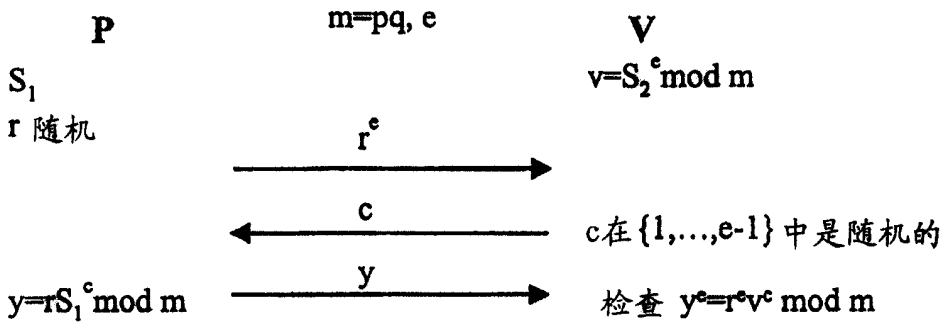

P
m=pq, e
V
S1
v=S2^e mod m
r 随机
r^e
c
c在{1,...,e-1}中是随机的
y
y=rS1^c mod m
检查 y^e=r^e v^c mod m

图 6

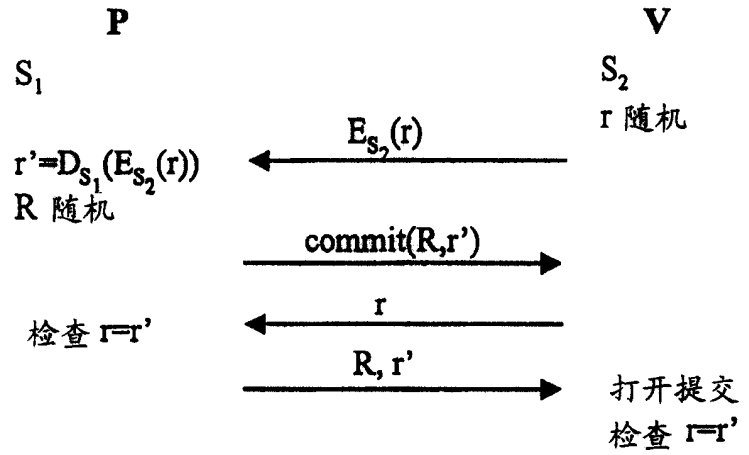

P
V
S1
S2
r 随机
E_S2(r)
r'=D_S1(E_S2(r))
R 随机
commit(R,r')
r
检查 r=r'
R, r'
打开提交
检查 r=r'

图 7