(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 103997407 A
(43)申请公布日 2014.08.20

(21)申请号 201410102426.8

(22)申请日 2014.02.17

(30)优先权数据
13305177.1 2013.02.15 EP
13305452.8 2013.04.08 EP

(71)申请人 汤姆逊许可公司
地址 法国伊西莱穆利诺

(72)发明人 M·乔伊 B·利伯特

(74)专利代理机构 北京市柳沈律师事务所 11105
代理人 吕晓章

(51)Int.Cl.
H04L 9/32(2006.01)
H04L 9/30(2006.01)
H04L 29/06(2006.01)

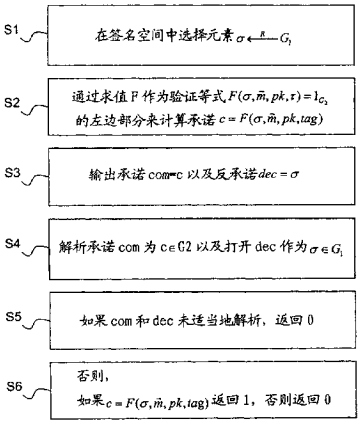
权利要求书1页 说明书11页 附图1页

(54)发明名称

用于生成和验证线性同态签名中的承诺的加密设备和方法

(57)摘要

设备(100)的处理器(120)通过以下步骤生成加密承诺:接收矢量$\vec{m}$、同态签名方案的公共验证密钥、以及标签;在签名空间中选择签名σ;通过运行该同态签名方案的验证算法生成承诺c;输出该承诺c作为来源于该验证算法的中间值。



CN 103997407 A

1. 一种生成不可延展的加密承诺的方法，该方法包括在设备 (110) 的处理器 (120) 中的步骤：

- 接收矢量$\vec{m}$、与签名存在的空间相关联的同态签名方案的公共验证密钥、以及标签；
- 在签名存在的空间选择元素 σ ；
- 使用该矢量$\vec{m}$、该公共验证密钥、该标签以及该元素 σ 生成承诺 c ；以及
- 输出该承诺 c ；

其中该承诺 c 通过在该矢量$\vec{m}$、该公共验证密钥、该标签以及该元素 σ 上运行该同态签名方案的验证算法被生成。

2. 根据权利要求 1 的方法，其中该承诺 c 的大小独立于该矢量$\vec{m}$的大小。

3. 根据权利要求 1 的方法，其中该矢量$\vec{m}$包括在其上双线性映射 $G \times G \rightarrow G_T$ 可被高效计算的组 G 的元素。

4. 根据权利要求 1 的方法，其中该矢量$\vec{m}$的维度大于或等于 2。

5. 根据权利要求 1 的方法，其中该承诺 c 允许使用零知识证明以证明打开的知识。

6. 根据权利要求 1 的方法，其中该承诺 c 被生成作为来源于该验证算法的中间值。

7. 一种生成不可延展的加密承诺的设备 (100)，该设备 (100) 包括：

- 至少一个界面 (110) 被配置为：
- 接收矢量$\vec{m}$、与签名存在的空间相关联的同态签名方案的公共验证密钥、以及标签；以及
- 输出该承诺 c ；以及
- 处理器 (120) 被配置为：
- 在签名存在的空间选择元素 σ ；以及
- 使用该矢量$\vec{m}$、该公共验证密钥、该标签以及该元素 σ 生成承诺 c ；

其中该处理器被配置为通过在该矢量$\vec{m}$、该公共验证密钥、该标签以及该元素 σ 上运行该同态签名方案的验证算法来生成承诺 c。

8. 根据权利要求 7 的设备，其中该承诺 c 的大小独立于该矢量$\vec{m}$的大小。

9. 根据权利要求 7 的设备，其中该矢量$\vec{m}$包括在其上双线性映射 $G \times G \rightarrow G_T$ 可被高效计算的组 G 的元素。

10. 根据权利要求 7 的设备，其中该矢量$\vec{m}$的维度大于或等于 2。

11. 根据权利要求 7 的设备，其中该承诺 c 允许使用零知识证明以证明打开的知识。

12. 根据权利要求 7 的设备，其中该承诺 c 被产生作为来源于该验证算法的中间值。

13. 一种非临时性的计算机程序产品 (140)，其储存当被处理器执行时执行权利要求 1-6 中任一所述方法的指令。

用于生成和验证线性同态签名中的承诺的加密设备和方法

技术领域

[0001] 本发明一般涉及加密，特别是线性同态签名中的不可延展的承诺。

背景技术

[0002] 本部分意在向读者介绍与以下被描述和/或要求保护的本发明的各个方面相关的技术的各个方面。相信本讨论有助于为读者提供背景信息以促进对本发明各个方面的更好的理解。相应地，应当理解的是这些叙述从这个角度被阅读，并不作为对现有技术的承认。

[0003] 所谓的承诺方案可被称作密封的信封的数字等价物：信封中的无论何物仍然保密直至该信封被打开。同时，一旦该信封已经被关闭，发送者不可改变他关于该内容的意见。因此该承诺方案的目标是促使发送者定义一个信息，其不能改变直至将来它被显示的那个时间。

[0004] 承诺方案可是非交互的，其意味着所谓的承诺阶段和打开(opening)阶段均包括从发送者到接收者的单一信息。换句话说，该接收者除了接收信息外不必以任何其他形式和发送者交互。

[0005] 陷门(trapdoor)承诺是完美的隐藏承诺(即，其中的隐藏性质甚至影响无限制的攻击者)，由于陷门tk使得打破绑定性质并打开承诺到任意值成为可能。然而，应当保持没有该陷门是不可实行的。如在本发明的描述中将被看到的，陷门承诺使用称作FakeCom和FakeOpen的两个额外算法。

[0006] 在仿真-声音陷门承诺(SSTC)中，每个承诺被用标签标记。在P.MacKenzie, K.Yang.On Simulation-Sound Trapdoor Commitments.In Eurocrypt'04,Lecture Notes in Computer Science,vol.3027,pages382-400,2004.中给出定义。该定义要求即使攻击者被允许看到承诺的等价物而可能区分多个标签$tag_1,\ldots,tag_q$的消息，也将无法打破该绑定性质用于新的标签$tag \notin \{tag_1,\ldots,tag_q\}$。

[0007] 承诺方案的另一个期望的性质是攻击者不能承诺与这些诚实的参与者相关的消息。关于打开的独立性的概念捕获到，攻击者可以对其打开承诺的消息应当独立于诚实的发送者的承诺被打开的方式。参见G.Di Grescenzo, Y.Ishai, R.Ostrovsky. Non-Interactive and Non-Malleable Commitment.In STOC'98, pages141-151,1998.以及R.Gennaro和S.Micali.Indepentdent Zero-Knowledge Sets.In ICALP'06, Lecture Notes in Computer Science, vol.4052, pages34-45,2006。

[0008] 对于承诺，素数阶$p>2^{\lambda}$的组(G,G_T)被考虑，其中λ为安全参数，在其上离散对数问题被假设为困难的。此外假定有效的可计算的双线性映射(又叫做，配对)；$e:G\times G\rightarrow G_T$。也就是，对于任一$g,h\in G$以及任一$a,b\in Z$，$e(g^a,h^b)=e(g,h)^{ab}$。此外，当且仅当$g\neq 1_G$和$h\neq 1_G$时，$e(g,h)\neq 1_{G_T}$。

[0009] 在这些组中，可能依赖于以下硬性假定：

[0010] • 计算的Diffie-Hellman问题(CDH)；

[0011] • 决策线性问题(DLIN)-参见D.Boneh,X.Boyen,H.Shacham.Short Group Signatures.In Crypto'04,Lecture Notes in Computer Science,vol.3152,pages41-55,Springer,2004;以及

[0012] .连续双配对问题(SDP)。这个假定通过组G中的DLIN假定被隐含-参见M.Abe,K.Haralambiev,M.Ohlcubo.Signing on Elements in Bilinear Groups for Modular Protocol Design.Cryptology ePrint Archive:Report 2010/133,2010。

[0013] 现有技术包括非交互不可延展承诺的多个构造,上述承诺不能被重复使用,因为在输出它自有的承诺前攻击者仅仅被给予一个诚实地生成的承诺。参见以下示例:

[0014] •G.Di Crescenzo,Y. Ishai,R.Ostrovsky.Non-Interactive and Non-Malleable Commitment.In STOC' 98,pp.141-150,1998.

[0015] • US6301664(G.Di Crescenzo,Y.Ishai,R.Ostrovsky.Method and System for Non-Malleable and Non-Interactive Cryptographic Commitment in a Network.2001)

[0016] • G.Di Crescenzo,J.Katz,R.Ostrovsky,A.Smith.Efficient and Non-interactive Non-malleable Commitment.In Eurocrypt' 01,Lecture Notes in Computer Science,vol.2045,pp.40-59,2001.

[0017] 可重复使用的不可延展的承诺的概念由Damgard和Groth提出[I.Damgard,J.Groth.Non-interactive and reusable non-malleable commitment schemes.In STOC' 03,pages426-437,2003.]。可重复使用的不可延展的承诺可从仿真-声音陷门承诺[参见J.Garay,P.MacKenzie,K.Yang.Strengthening Zero-Knowledge Protocols Using Signatures.In Eurocrypt' 03,Lecture Notes in Computer Science,vol.2656,pp.177-194,2003. 以 及 P.MacKenzic,K.Yang.On Simul ation-Sound Trapdoor Commitments.In Eurocrypt ' 04,Lecture Notes in Computer Science,vol.3027,pp.382-400,2004.]和多陷门承诺[R.Gemiaro.Multi-Trapdoor Commitments and Their Applications to Proofs of Knowledge Secure Under Concurrent Man-in-the-Middle Attacks.In Crypto' 04,Lecture Notes in Computer Science,vol.3152,pp.220-236,2004]中被构造。

[0018] 这些论文中,MacKenzie和Yang从签名方案承认效率Σ协议中给出SSTC的架构。如在现有技术中众所周知的,Σ协议是证人和验证者之间的三运动交互协议。事实上,如Fujisaki指出的[E.Fuiisaki.New Constructions of Efficient Simulation-Sound Commitments Using Encryption and Their Applications.In CT-RSA ' 12,Lecture Notes in Computer Science.vol.7178,pp.136-155,2012.],建立在签名方案上的非交互仿真-声音或者多陷门[参见Gennaro的论文]承诺的所有已知构造,效率Σ协议允许签名的证明知识。

[0019] 想法是通过使用m作为对Σ协议的挑战(challenge)用来在标签上证明签名σ=Sig(sk,tag)的知识来承诺一个消息m。该承诺通过Σ协议的副本(a,m,z)的第一消息a被给出,其通过仿真消息tag上的有效签名σ的知识的证明来获取。该承诺随后通过揭示z被打开。通过该Σ协议的特定的可靠性,除非发送者实际上知道tag上有效的签名,它只能打开给定的承诺a到一个消息m。

[0020] 而简单的,如果我们希望发送者仍然能够高效地证明关于承诺的矢量的独立坐标

的声明，上面的构造并不容易地扩展到对矢量的承诺。此外，基于Σ协议的方法并不能使得可以承诺组元素的矢量：为了这个目的，我们将需要在其中挑战是组元素的矢量的Σ协议。在之前提到的论文中，Fujiisaki给出了基于加密方案的SSTC系统的可替代的构造。然而，这个构造是交互的，因为在承诺阶段需要发送者和接收者之间的多个交互的循环。

[0021] 组元素的非交互承诺被描述在例如：

[0022] • J.Groth.Simulation-Sound NIZK Proofs for a Practical Language and Constant Size Group Signatures.In Asiacrypt′ 06，Leclure Notes in Computer Science，vol.4284，pp.444-459，Springer，2006.

[0023] • J.Groth，A.Sahai.Efficient non-interactive proof systemsfor bilinear groups.In Eurocrypt′ 08，Lecture Notes in Computer Science，vol.4965，pp.415-432，2008.

[0024] • J.Cathalo，B.Liberr，M.Yung.Group Encryption：Non-Interactive Realization in the Standard Model.In A.siacryht′ 09，Lecture Notes in Computer Science，vol.5912，pp.179-196，2009.

[0025] •J.Groth.Homomorphic trapdoor commitments to group elements.Cryptology ePrint Archive：Report2009/007，2009.

[0026] 然而，这些方案都是同态的并且因此可延展。迄今为止，唯一知道的其消息和打开仅仅包含组元素的非交互不可延展承诺方案由Fischlin等描述［参见M.Fischlin，B.Libcrt，M.Manulis.Non-interactive and Re-usable Universally Composable String Comnitments with Adaptive Security.In Asiacrypt’11，Lecture Notes in Computer Science，vol.7073，pp.468-485，2011.］。然而，这些构造不能减少长度（即，承诺不能短于消息），因为他们达到通用可组合性［参见R.Canetti.Universally Composable Security：A New Paradigm for Cryptographic Protocols.In FOCS′ 01pp.136-145，2001. 以 及 R.Canetti，M.Fischlin.Universally Composable Commitments.In Crypto’01，Lecture Notes in Computer Sciece，vol.2139，pp.19-40，2001.］并且已知的是，通用可组合的承诺必须是可抽取出的（也就是，陷门信息应当使得可能恢复承诺中包含的消息）。

[0027] 因而应当理解的是，期望具有一个承诺方案，其提供对矢量$\vec{m}=(m_1,\ldots,m_n)$的非交互不可延展陷门承诺的模块化构造，同时保持高效证明关于单个矢量坐标$\{m_i\}_{i=1}^n$的属性的能力。这排除了包含对矢量$\vec{m}$的哈希值的承诺的无价值的方案。还需要该方案能够对组元素承诺而无需知道它们的离散对数。换句话说，要被承诺的消息应包括组元素的矢量$(M_1,\ldots,M_n)\in G^n$，其中G是一个组，在其中双线性映射$e:G\times G\rightarrow G_T$是高效的可计算的。该承诺方案也应优选地被设计以使承诺字串com具有固定大小，无论一次承诺了多少组元素$(M_1,\ldots,M_n)$。最后，打开也应优选地包括G中的元素，其将使得能够生成承诺的组元素满足特定的性质的高效的非交互的证明（使用[J.Groth，A.Sahai.Efficient non-interactive proof systems for bilinear groups.In Eurocrypt’08，Lecture Notes in Computer Science，vol.4965，pp.415-432，2008.］的技术）。本发明提供这样的承诺方案。

发明内容

[0028] 在第一方面，本发明针对一种生成不可延展的加密承诺的方法。设备的处理器接收矢量、与签名存在的空间相关联的同态签名方案的公共验证密钥、以及标签；在签名存在的空间选择元素；使用该矢量、该公共验证密钥、该标签以及该元素生成承诺；以及输出该承诺。该承诺 c 通过在该矢量、该公共验证密钥、该标签以及该元素上运行该同态签名方案的验证算法被生成。

[0029] 在第一实施例中，该承诺的大小独立于该矢量的大小。

[0030] 在第二实施例中，其中该矢量包括在其上双线性映射 $G \times G \rightarrow G_T$ 可被高效计算的组 G 的元素。

[0031] 在第三实施例中，其中该矢量的维度大于或等于 2。

[0032] 在第四实施例中，该承诺允许使用零知识证明以证明打开的知识。

[0033] 在第五实施例中，该承诺被生成作为来源于该验证算法的中间值。

[0034] 在第二方面，本发明针对一种生成不可延展的加密承诺的设备，该设备包括至少一个界面，被配置为：接收矢量、与签名存在的空间相关联的同态签名方案的公共验证密钥、以及标签；以及输出该承诺。该设备进一步还包括一处理器，被配置为：在签名存在的空间选择元素；以及使用该矢量、该公共验证密钥、该标签以及该元素生成承诺。该处理器被配置为通过在该矢量、该公共验证密钥、该标签以及该元素上运行该同态签名方案的验证算法生成承诺。

[0035] 在第一实施例中，该承诺的大小独立于该矢量的大小。

[0036] 在第二实施例中，其中该矢量包括在其上双线性映射 $G \times G \rightarrow G_T$ 可被高效计算的组 G 的元素。

[0037] 在第二实施例中，其中该矢量的维度大于或等于 2。

[0038] 在第三实施例中，该承诺允许使用零知识证明以证明打开的知识。

[0039] 在第四实施例中，该承诺被生成作为来源于该验证算法的中间值。

[0040] 在第三方面，本发明针对一种非临时性的计算机程序产品，其储存当被处理器执行时执行第一方面的任一实施例中的方法的指令。

附图说明

[0041] 现在将通过非限制性的示例的方式，参考附图描述本发明优选的特征，附图中：

[0042] 图 1 示出了根据本发明优选的实施例的一种用于生成承诺的加密设备以及一种用于验证承诺的加密设备；以及

[0043] 图 2 示出了根据本发明优选的实施例的一种用于生成承诺并用于验证承诺的打开的方法。

具体实施方式

[0044] 本发明的基本理念基于，在特定的温和条件下，线性同态结构保留的签名暗示对组元素矢量的长度减少不可延展结构保留的承诺。因此，本发明提供了长度减少不可延展结构保留的陷门承诺。将注意的是，该方案不是严格的结构保留（就是说承诺字串并不和消息存在于相同的组，根据 M. Abe, K. Haralambiev, M. Ohkubo. Group to Group Commitments Do Not Shrink. In Eurocrypt' 12, Lecture Notes in Computer Science, vol. 7237,

pp.301-317,2012. 的术语)。相反,该方案是非严格意义上的结构保留,因为承诺字串存在于 G_T 而不是 G(但是,如论文中所示的,严格的结构保留承诺不能是长度减少的)。仍然,打开仅仅包括 G 中的元素,其使得能够生成承诺的组元素满足特定特性的高效非交互证明。

[0045] 本发明的方案通过对组元素的第一构造仿真-声音陷门承诺(SSTC)被获取(参见 J.Garay,P.MacKenzie,K.Yang.Strengthening Zero-Knowledge Protocols Using Signatures.In Eurocrypt ′ 03,Lecture Notes in Computer Science,vol.2656,pp.177-194,2003 以及 P.MacKenzie,K.Yang.On Simulation-Sound Trapdoor Commitments.In Eurocrypt ′ 04,Lecture Notes in Computer Science,vol.3027,pp.382-400,2004)。

[0046] 在Garay、MacKenzie Yang的论文中SSTC方案最先被建议作为工具用于构造通用可组合零知识证明(参见R.Canetti.Universally Composable Security:A New Paradigm for Cryptographic Protocols.In FOCS′ 01,pp.136-145,2001.)。MacKenzie 以及 Yang 随后给出了简化的安全定义,其足够提供从可重复使用的不可延展承诺的定义的意义上的与打开相关的不可延展性(参见 I.Damgard,J.Groth.Non-interactive and reusable non-malleable commitment schemes.In STOC'03,pages426-437,2003.)。

[0047] 首先指出的是任何固定长度的线性同态结构保留的签名需要遵从以下模板。

[0048] Keygen(pp,n):给定的公共参数 pp,其包括具有双线性映射的组 (G,G_T) 的描述,以及要被签名的子空间的维度 $n \in N$,选择常数 n_z、n_v、$m \in N$。其中,n_z 和 n_v 决定了签名的长度,m 为验证算法中方程式的数目。然后在组 G 中选择元素 $\{F_j,\mu\}_{j\in\{1,\ldots,m\},\mu\in\{1,\ldots,nz\}}$,$\{G_{ji}\}_{i\in\{1,\ldots,n\},j\in\{1,\ldots,m\}}$。该公共密钥为

[0049] $$pk=(\{F_j,\mu\}_{j\in\{1,\ldots,m\},\mu\in\{1,\ldots,n_z\}},\{G_{ji}\}_{i\in\{1,\ldots,n\},j\in\{1,\ldots,m\}})$$

[0050] 并且该私人密钥包括与涉及特定基础的公共密钥元素的表示相关的信息。

[0051] Sign(sk,τ,$(M_1,\ldots,M_n)$):输出元组

[0052] $$\sigma=(Z_1,\ldots Z_{n_z},V_1,\ldots V_{n_v})\in G^{n_z+n_v}$$

[0053] $\text{SignDerive}(pk,\tau,\{\omega_i,\sigma^{(i)}\}_{i=1}^{\ell})$:对于 i = 1 到 l,解析每个签名 $\sigma^{(i)}$ 作为 $(Z_1^{(i)},\ldots,Z_{n_z}^{(i)},V_1^{(i)},\ldots,V_{n_v}^{(i)})$,然后计算

[0054] $$Z_\mu=\prod_{i=1}^{\ell}Z_\mu^{(i)^{\omega_i}}\quad V_v=\prod_{i=1}^{\ell}V_v^{(i)^{\omega_i}}\qquad \mu\in\{1,\ldots,n_z\},v\in\{1,\ldots,n_v\}$$

[0055] 以及,可能在重新随机分布步骤后,输出 $\sigma=(Z_1,\ldots Z_{n_z},V_1,\ldots V_{n_v})$。

[0056] Verify(pk,σ,τ,$(M_1,\ldots,M_n)$):给定签名 $\sigma=(Z_1,\ldots Z_{n_z},V_1,\ldots V_{n_v})\in G^{n_z+n_v}$,矢量 $(M_1,\ldots,M_n)$ 以及标签 τ,如果 $(M_1,\ldots,M_n)=(1_G,\ldots,1_G)$ 返回 0,否则:

[0057] 1. 对于每个 $j\in\{1,\ldots,m\}$ 以及 $v\in\{1,\ldots,n_v\}$,计算标签 τ 的一对一的编码 $T_{j,v}\in G$ 作为组元素。(可能松弛这个条件以具有耐碰撞确定性编码,这里为了简化假定内射(injectivity))

[0058] 1. 对于 j = 1 到 m,计算

[0059] $$c_j = \prod_{\mu=1}^{n_z} e\left(F_{j,\mu}, Z_\mu\right) \cdot \prod_{v=1}^{n_v} e\left(T_{j,v}, V_v\right) \cdot \prod_{i=1}^{n} e\left(G_{j,i}, M_i\right)$$

[0060] 2. 对于每一个 $j \in \{1,\ldots,m\}$，当且仅当$c_j = 1_{G_T}$时返回 1。

[0061] 在以下的描述中，如果对于每一个文件识别符（例如“标签”），任何非不重要矢量 $(M_1,\ldots,M_n) \neq (1_G,\ldots,1_G)$ 有一个验证签名，则线性同态结构保留签名为“规则的”。

[0062] 图 1 示出了根据本发明优选的实施例的一种用于生成承诺的加密设备 100 以及一种用于验证承诺的加密设备 200。该设备 100、200 的每一个包括至少一个被配置用来通信的界面单元 110、210，至少一个处理器（“处理器”）120、220 以及至少一个诸如累加器的存储器 130、230，被配置用来存储诸如中间计算结果的数据和。图 1 也示出了诸如 CD-ROM 或 DVD 的第一和第二计算机程序产品（非临时性存储介质）140、240，其包含存储的指令，当该指令被处理器 120，220 执行时相应地生成以及验证根据本发明的承诺。

[0063] 结构保留仿真 - 声音陷门承诺的构造：

[0064] 如果 Π^{SPS} = (Keygen, Sing, Sign Derive, Verify) 是线性同态结构保留签名 (SPS)，则结构保留仿真 - 声音陷门承诺 (SSTC) 可被构造如下：

[0065] SSTC.Setup(λ，n)：给定承诺的矢量的期望维度 $n \in N$，为线性同态 SPS 方案选择公共参数 pp。然后对于常量 n_z、n_v、m 以及密钥 sk 运算 Π^{SPS} •Keygen(pp, n) 以获取 pk($\{F_j, \mu\}_{j\in\{1,\ldots,m\},\mu\in\{1,\ldots,nz\}}$，$\{G_{ji}\}_{i\in\{1,\ldots n\},j\in\{1,\ldots,m\}}$)。该承诺密钥为 pk=pk 并且陷门 tk 包括 sk。需要注意的是该公共密钥为常量 n_z 和 n_v 定义一签名空间$G^{n_z+n_v}$。

[0066] SSTC.Com(pk, tag, $(M_1,\ldots,M_n)$)：为了关于标签 tag = τ 承诺矢量 $(M_1,\ldots,M_n) \in G^n$，在签名空间中选择$\left(Z_1,\ldots,Z_{n_z},V_1,\ldots,V_{n_v}\right) \xleftarrow{R} \mathbb{G}^{n_z+n_v}$，然后运行 Verify(pk，σ，τ，$(M_1,\ldots,M_n)$) 的步骤 1 和 2，即，计算

[0067] $$c_j = \prod_{\mu=1}^{n_z} e(F_{j,\mu}, Z_\mu) \cdot \prod_{v=1}^{n_v} e(T_{j,v}, V_v) \cdot \prod_{i=1}^{n} e(G_{j,i}, M_i)$$

[0068] $j \in \{1,\ldots,m\}$

[0069] 这里 $\{T_{j,v}\}_{j,v}$ 形成 tag = τ 的内射编码作为组元素集合。该承诺字串由 com = $(c_1,\ldots,c_m)$ 给出，然而反承诺包含$\mathrm{dec} = \left(Z_1,\ldots,Z_{n_z},V_1,\ldots,V_{n_v}\right)$。

[0070] SSTC.FakeCom(pk，tk，tag)：利用 $(\hat{M}_1,\ldots,\hat{M}_n) \xleftarrow{R} G^n$ 像 SSTC.Com 一样进行。如果$(\hat{com},\hat{dec})$表示结果对，该算法输出$\tilde{com} = \hat{com}$以及 aux，对于 tag = τ 其包括对 aux $aux = ((\hat{M}_1,\ldots,\hat{M}_n),\hat{dec})$。

[0071] $\mathrm{SSTC.FakeOpen}\left(\mathrm{aux}, tk, tag, \tilde{\mathrm{com}}, (M_1,\ldots,M_n)\right)$：解析$\tilde{com}$为$(\tilde{c}_1,\ldots,\tilde{c}_n)$以及解析 aux 为$((\hat{M}_1,\ldots,\hat{M}_n),(\hat{Z}_1,\ldots,\hat{Z}_{n_z},\hat{V}_1,\ldots,\hat{V}_{n_v}))$。它首先在$(M_1/\hat{M}_1,\ldots,M_n/\hat{M}_n)$上对于 tag = τ 生成线性同态签名。也就是，使用陷门 tk = sk，计算

[0072] $$\sigma' = (Z'_1,\ldots Z'_{n_z},V'_1,\ldots V'_{n_v}) \leftarrow \Pi^{SPS} \cdot Sign(sk,\tau,((M_1/\hat{M}_1,\ldots,M_n/\hat{M}_n))$$

[0073] 因为 σ′ 是有效签名并且$\mathrm{aux}=((\hat{M}_1,\ldots,\hat{M}_n),(\hat{Z}_1,\ldots,\hat{Z}_{n_z},\hat{V}_1,\ldots,\hat{V}_{n_v}))$满足

[0074] $$\tilde{c}_j = \prod_{\mu=1}^{n_z} e(F_{j,\mu}, \hat{Z}_\mu) \cdot \prod_{v=1}^{n_v} e(T_{j,v}, \hat{V}_v) \cdot \prod_{i=1}^{n} e(G_{j,i}, \hat{M}_i) \qquad j \in \{1,\ldots,m\}$$

[0075] FakeOpen 算法可运行

[0076] $$(\tilde{Z}_1,\ldots,\tilde{Z}_{n_z},\tilde{V}_1,\ldots,\tilde{V}_{n_v}) \leftarrow Sign.Derive(pk,\tau,\{(1,\sigma'),(1,\hat{\sigma})\}),$$

[0077] 其中$\sigma' = (\hat{Z}_1,\ldots\hat{Z}_{n_z},\hat{V}_1,\ldots\hat{V}_{n_v})$。通过构造，$d\hat{e}c = (\tilde{Z}_1,\ldots,\tilde{Z}_{n_z},\tilde{V}_1,\ldots,\tilde{V}_{n_v})$是关于 tag = τ 对矢量 $(M_1,\ldots,M_n)$ 的有效的反承诺。

[0078] SSTC.Verify(pk, tag, $(M_1,\ldots,M_n)$, com, dec)：解析该承诺为$(c_1,\ldots,c_m) \in G_T^m$并且打开 dec 为$(Z_1,\ldots Z_{n_z},V_1,\ldots V_{n_v}) \in G^{n_z+n_v}$。如果这些没有被正确地解析，返回 0。然后计算 tag = τ 的一对一编码 $\{T_{j,v}\}_{j,v}$。如果保持

[0079] $$c_j = \prod_{\mu=1}^{n_z} e(F_{j,\mu}, Z_\mu) \cdot \prod_{v=1}^{n_v} e(T_{j,v}, V_v) \cdot \prod_{i=1}^{n} e(G_{j,i}, M_i) \qquad j \in \{1,\ldots,m\}$$

[0080] 则返回 1，否则返回 0。

[0081] 来自线性同态签名的 SSTC

[0082] 本部分提供了对前面结构保留构造的概括。其目标是从线性同态签名构造对矢量的仿真 - 声音（因而不可延展）承诺。要注意的是用于构造 SSTC 的现有技术方案并不能直接允许对矢量承诺的同时保留对承诺的矢量的知识的高效证明可行性。该方法被示出在图 2 中。

[0083] 使Π = (Keygen, Sign, Sign Derive, Verify) 为Z_p^n上一些大的素数 $p>2^\lambda$ 的线性同态签名。假定对于一些 k ∈ N，Π分别使用公共阶 p^k 和 p 的组 G_1 和 G_2。也假定每一签名 σ 存在于 G_1 上。该验证算法采用声称的签名 σ ∈ G_1，文件识别符 τ 以及矢量$\vec{m}$作为输入。当且仅当$F(\sigma,\vec{m},pk,\tau) = 1_{G_2}$时返回 1，这里 F 是一个涉及组 G_2 并满足特定线性性质的函数。也就是，对于 Keygen 产生的每一个 pk 和每一 τ，要求对于任意矢量$\vec{m}_1,\vec{m}_2 \in Z_p^n$以及任意 σ_1，$\sigma_2 \in G_1$，$F(\sigma_1\cdot\sigma_2,\vec{m}_1+\vec{m}_2,pk,\tau) = F(\sigma_1,\vec{m}_1,pk,\tau)\cdot F(\sigma_2,\vec{m}_2,pk,\tau)$。因而，对于任意 ω ∈ Z_p 和任意 σ ∈ G_1，$F(\sigma,\vec{m},pk,\tau)^\omega = F(\sigma^\omega,\vec{m},pk,\tau)$。

[0084] 应注意的是该模板仅捕获公共阶的组的方案，从而使基于强 RSA 假定的构造不被包括在内。其原因在于，当在整数中工作时，消息和签名组件可能会在每次同态操作处增加，这使得很难呈现不能从原始的反承诺中区分的伪造的打开。

[0085] SSTC.Setup(λ, n)：给定承诺的矢量的期望维度 n ∈ N，为该线性同态签名选择公共参数 pp。然后运行密钥生成算法Π^{SPS}•Keygen(pp, n) 以获得公共密钥 pk 以及私人密钥 sk。该承诺密钥为 pk = pk 以及该陷门 tk 包括 sk。

[0086] SSTC.Com $(pk,tag,\vec{m})$：为了关于 tag 承诺矢量$\vec{m} \in Z_p^n$，在签名空间中选择 S1 元素$\sigma \xleftarrow{R} G_1$。通过求值 F 作为该验证等式$F(\sigma,\vec{m},pk,\tau) = 1_{G_2}$的左边部分计算 S2 并输出 S3 $c = F(\sigma,\vec{m},pk,tag)$。该承诺字串为 com = c，而该反承诺为 dec = σ。

[0087] SSTC.FakeCom(pk, tk, tag)：像 SSTC.Com 那样进行，但使用随机选择的矢量$\vec{m}_{fake} \xleftarrow{R} Z_p^n$，如果$(c\hat{o}m, d\hat{e}c)$表示结果的承诺 / 反承诺对，该算法设置$c\tilde{o}m = c\hat{o}m$以及

$aux=(\vec{m}_{fake},d\hat{e}c)$。

[0088] SSTC.FakeOpen $(aux,tk,tag,c\tilde{o}m,\vec{m})$：解析 $c\tilde{o}m$ 为 $\tilde{c}\in G_2$ 并解析 aux 为 $(\vec{m}_{fake},d\hat{e}c)$，其中 $d\hat{e}c=\hat{\sigma}\in G_1$。首先为 tag = τ 在坐标差 $\vec{m}-\vec{m}_{fake}\in Z_p^n$ 上生成线性同态签名。也就是，使用陷门 tk = sk，计算 $\sigma'\leftarrow\prod.Sign(sk,\tau,\vec{m}-\vec{m}_{fake})$。最后，计算 $\tilde{\sigma}=\hat{\sigma}\cdot\sigma'\in G_1$ 并返回 $d\tilde{e}c=\tilde{\sigma}$。

[0089] SSTC.Verify: $(pk,tag,\vec{m},com,dec)$：解析 S4 该承诺 com 为 c ∈ G2 以及打开 dec 为 σ ∈ G_1。如果这些未被正确地解析，返回 S50。否则，如果 $c=F(\sigma,\vec{m},pk,tak)$，返回 S61，否则返回 0。

[0090] Attrapadung 等人的线性同态方案（参见 N. Attrapadung, B. Libert, T. Peters. Computing on Authenticated Data：New Privacy Definitions and Constructions. In Asiacrypt′ 12, in Lecture Notes in Computer Science vol. 7658, pp. 367-385, 2012.）可被看作该模板的特定实例，其中组 G_1 为一乘积 $G_1=G^2\times Z_p$，其是用于操作 $(\cdot,\cdot,+)$ 的组并且 $G_2=G_T$。在该方案中，G_1 和 G_2 因而分别具有阶 p^3 和 p。对于线性函数 F，它可被实例化为

[0091] $$F((\sigma_1,\sigma_2,s),\vec{m},pk,\tau):=e(\sigma_1,g^{-1})\cdot e(H_G(\tau),\sigma_2)\cdot e(g_1^{m_1}\cdots g_n^{m_n}\cdot v^s,g^a)$$

[0092] 因此，获得对于基于计算的 Diffie-Hellman(CDH) 的矢量的新的非交互仿真 - 声音陷门承诺。

[0093] 应注意的是这个方案可通过移除项 v^s 和 s 被优化以使

[0094] $(\sigma_1,\sigma_2)=((\prod_{i=1}^n g_i^{m_i})^\alpha\cdot H_G(\tau)^r,g^r)$ 以及

[0095] $$F((\sigma_1,\sigma_2),\vec{m},pk,\tau):=e(\sigma_1,g^{-1})\cdot e(H_G(\tau),\sigma_2)\cdot e(g_1^{m_1}\cdots g_n^{m_n},g^a)$$

[0096] 事实上，虽然该项 $(v^s,s)\in G\times Z_p$ 在以下签名方案中是必需的，但是他们可在结果的承诺中被消除。

[0097] 优化善于应用在以下 SSTC 方案中，其依赖 CDH 假定并允许对矢量的承诺。但应注意的是基于 CDH 假定的现有技术的不可延展承诺被 - 或明示或暗示地 - 描述在两篇论文中，但还不清楚如何以模块化的方式扩展他们以对矢量承诺（参见 Y. Dodis, V. Shoup, S. Walfish. Efficient Constructions of Composable Commitments and Zero-Knowledge Proofs. In Crypfo′ 08, Lecture Notes in Computer Science, vol. 5157, pp. 21-38, 2008. 以及 R. Nishimaki, E. Fujisaki, K. Tanaka. A Multi-trapdoor Commitment Scheme from the RSA Assumption. In ACISP2010, Lecture Notes in Computer Science, vol. 6168, pp. 182-199, 2010.）

[0098] SSTC.Setup(λ, n)：给定承诺的矢量的期望维度 n ∈ N，选择素数阶 $p>2^\lambda$ 的双线性组 (G, GT)。对于一些 L ∈ poly(λ)，选择 $\alpha\xleftarrow{R}Z_p,g\xleftarrow{R}G$ 以及 $u_0,\ldots,u_L\xleftarrow{R}G$。这些元素 $u_0,\ldots,u_L\in G^{L+1}$ 被用来执行数论哈希函数 $H_G:\{0,1\}^L\rightarrow G$ 以使任一 L 位字符 τ = τ[1]... τ[L] ∈ $\{0,1\}^L$ 被映射到哈希值 $H_G(\tau)=u_0\cdot\prod_{i=1}^L u_i^{\tau[i]}$。然后为 i = 1 到 n 选择 $g_i\xleftarrow{R}G$，并定义识别符空间 Γ := $\{0,1\}^L$。该陷门为 tk = α 并且该公共密钥包含 $pk=((G,G_T),g,g^\alpha,\{g_i\}_{i=1}^n,\{u_i\}_{i=1}^L)$。

[0099] $\mathbf{SSTC.Com}\ (pk,tag,\vec{m})$：为了关于 tag 承诺矢量$\vec{m}=(m_1,...,m_n)\in Z_p^n$，在签名空间中选择元素$\sigma_1,\sigma_2\xleftarrow{R}G$并计算

[0100] $c=e(g_1^{m_1}\cdots g_n^{m_n},g^a)\cdot e(g^{-1},\sigma_1)\cdot e(H_G(tag),\sigma_2)$。

[0101] 返回承诺字串 com = c ∈ G_T 以及反承诺 dec = (σ 1, σ 2) ∈ G^2。

[0102] SSTC.FakeCom(pk, tk, tag)：像 SSTC.Com 一样进行但是使用随机选择矢量$\vec{m}_{fake}\xleftarrow{R}Z_p^n$。如果$(\hat{com},\hat{dec})$表示结果承诺/反承诺对，该算法设置$\tilde{com}=\hat{com}$以及$aux=(\vec{m}_{fake},\hat{dec})$。

[0103] $\mathbf{SSTC.FakeOpen}\ (aux,tk,tag,\tilde{com},\vec{m})$：解析$\tilde{com}$为$\tilde{c}\in G_T$以及解析 aux 为$(\vec{m}_{fake},\hat{dec})$，这里$\hat{dec}=(\sigma_1,\sigma_2)\in G^2$。首先为标签 tag 在差$(m_1',...,m_n')=\vec{m}-\vec{m}_{fake}\in Z_p^n$上生成线性同态签名（σ′ 1, σ′ 2）∈ G^2。也就是，使用 tk = α，计算$\sigma_1'=(g_1^{m_1'}\cdots g_n^{m_n'})^{\alpha}\cdot H_G(tag)^r$以及 σ′ 2 = g^r 其满足

[0104] $1_{G_T}=e(g_1^{m_1'}\cdots g_n^{m_n'},g^a)\cdot e(g^{-1},\sigma_1')\cdot e(H_G(tag),\sigma'_2)$

[0105] 最后，计算$(\tilde{\sigma}_1,\tilde{\sigma}_2)=(\hat{\sigma}_1\cdot\sigma_1',\hat{\sigma}_2\cdot\sigma_2')\in G^2$以及返回$\tilde{dec}=(\tilde{\sigma}_1,\tilde{\sigma}_2)$。

[0106] $\mathbf{SSTC.Verify}(pk,tag,\vec{m},com,dec)$：解析该承诺 com 为 c ∈ G_T 以及打开 dec 为（σ 1, σ 2）∈ G^2。如果这些未被正确地解析，返回 0。否则，如果（σ 1, σ 2）满足$c=e(g_1^{m_1}\cdots g_n^{m_n},g^a)\cdot e(g^{-1},\sigma_1)\cdot e(H_G(tag),\sigma_2)$返回 1，否则返回 0。

[0107] 在表示进一步的 SSTC 方案之前，这里遵从它的以下线性同态结构保留签名方案的描述。

[0108] Keygen(pp, n)：给定安全参数 λ 以及要被签名的子空间的维度 n ∈ N，选择素数阶 p>2^{λ} 的双线性组 (G, G_T)，执行如下步骤：

[0109] 1. 选择$h\xleftarrow{R}G$以及$\alpha_z,\alpha_r,\beta_z\xleftarrow{R}Z_p$，定义$g_z=h^{\alpha_z}$，$g_r=h^{\alpha_r}$以及$h_z=h^{\beta_r}$。

[0110] 2. 对于每一个 i ∈ {1, ..., n}，挑选$\chi_i,\gamma_i,\delta_i\xleftarrow{R}Z_p$并计算$g_i=g_z^{\chi_i}g_r^{\gamma_i}$，$h_i=h_z^{\chi_i}h_r^{\delta_i}$。

[0111] 3. 选择随机矢量$\vec{w}=(w_0,w_1,...,w_L)\xleftarrow{R}G^{L+1}$，其定义了哈希函数 H_G：{0,1}^{L} → G，其映射 τ = τ[1]... τ[L] ∈ {0,1}^{L} 到$H_G(\tau)=w_0\cdot\prod_{K=1}^{L}w_k^{\tau[k]}$。

[0112] 该公共密钥包含

[0113] $pk=(g_Z,g_r,h_Z,h,\{g_i,h_i\}_{i=1}^n,\vec{w})\in G^{2n+4}\times G^{L+1}$

[0114] 且该私人密钥为$sk=(h_z^{\alpha_r},(\chi_i,\gamma_i,\delta_i)_{i=1}^n)$。

[0115] Sign(sk, τ, (M_1, ..., M_n))：为了相对于文件识别符 τ 使用$sk=(h_z^{\alpha_r},(\chi_i,\gamma_i,\delta_i)_{i=1}^n)$签名矢量 ($M_1$, ..., M_n) ∈ G^n，选择$\theta,\rho\xleftarrow{R}Z_P$并计算

[0116] $z=g_r^{\theta}\cdot\prod_{i=1}^{n}M_i^{-\chi_i}$，　$r=g_Z^{-\theta}\cdot\prod_{i=1}^{n}M_i^{-\gamma_i}$，　$u=(h_Z^{\alpha_r})^{-\theta}\cdot\prod_{i=1}^{n}M_i^{-\delta_i}\cdot H_G(\tau)^{-\rho}$，

[0117] v = h^{ρ}

[0118] 该签名包括 $\sigma = (z, r, u, v) \in G^4$。

[0119] $\mathrm{SignDerive}(pk, \tau, \{\omega_i, \sigma^{(i)}\}_{i=1}^{\ell})$：给定 pk，文件识别符 τ 以及 l 元组 $(\omega_i, \sigma^{(i)})$，对于 i = 1 到 l 解析每一个签名 $\sigma^{(i)}$ 为 $\sigma^{(i)} = (z_i, r_i, u_i, v_i) \in G^4$。然后选择$\rho' \xleftarrow{R} Z_p$并计算

[0120] $$z = \prod_{i=1}^{\ell} z_i^{\omega_i} \quad r = \prod_{i=1}^{\ell} r_i^{\omega_i} \quad u = \prod_{i=1}^{\ell} u_i^{\omega_i} \cdot H_{\mathbb{G}}(\tau)^{-\rho'} \quad v = \prod_{i=1}^{\ell} v_i^{\omega_i} \cdot h^{\rho'}$$

[0121] 并返回 $\sigma = (z, r, u, v)$。

[0122] Verify(pk, σ, τ, $(M_1, \ldots, M_n)$)：给定签名 $\sigma = (z, r, u, v) \in G^4$，文件识别符 τ 以及矢量 $(M_1, \ldots, M_n)$，当且仅当 $(M_1, \ldots, M_n) \neq (1_G, \ldots, 1_G)$ 以及 (z, r, u, v) 满足等式

[0123] $$1_{G_T} = e(g_z, z) \cdot e(g_r, r) \cdot \prod_{i=1}^{n} e(g_i, M_i)$$

[0124] $$1_{\mathbb{G}_T} = e(h_z, z) \cdot e(h, u) \cdot e(H_{\mathbb{G}}(\tau), v) \cdot \prod_{i=1}^{n} e(h_i, M_i).$$

[0125] 时返回 1。

[0126] 来自线性同态结构保留签名的 SSTC

[0127] 一个特别有利的实施例是通过应用结构保留仿真 - 声音陷门承诺的构造到下文描述的线性同态签名获得的。

[0128] 类似下面的同态签名，结果 SSTC（其是结构保留的）的安全性依赖于连续双配对 (SDP) 问题的困难度。

[0129] SSTC.Setup(λ, n)：给定承诺的矢量的期望维度 $n \in N$，选择素数阶 $p > 2^{\lambda}$ 的双线性组 (G, G_T)。然后：

[0130] 1. 选择$h \xleftarrow{R} G$以及$\alpha_z, \alpha_r, \beta_z \xleftarrow{R} Z_p$。

[0131] 2. 定义$g_z = h^{\alpha_z}, g_r = h^{\alpha_r}$以及 $h_z = h^{\beta_z}$。

[0132] 3. 对于每一个 $i \in \{1, \ldots, n\}$，挑选 $\chi_i, \gamma_i, \delta_i \xleftarrow{R} Z_p$ 并计算 $g_i = g_z^{\chi_i} g_r^{\gamma_i}, h_i = h_z^{\chi_i} h_r^{\delta_i}$。

[0133] 4. 选择随机矢量$\vec{w} = (w_0, \ldots, w_L) \xleftarrow{R} \mathbb{G}^{L+1}$，其定义了哈希函数 $H_G: \{0,1\}^L \to G$，其映射 $\tau = \tau[1] \ldots \tau[L] \in \{0,1\}^L$ 到$H_G(\tau) = w_0 \cdot \prod_{K=1}^{L} w_k^{\tau[k]}$。

[0134] 该公共密钥包括$pk = (g_Z, g_r, h_Z, h, \{g_i, h_i\}_{i=1}^{n}, \vec{w}) \in G^{2n+4} \times G^{L+1}$

[0135] 同时该陷门为$tk = (h_z^{\alpha_r}, (\chi_i, \gamma_i, \delta_i)_{i=1}^{n})$。

[0136] SSTC.Com(pk, tag, $M_1, \ldots, M_n$))：为了承诺矢量 $(M_1, \ldots, M_n) \in G^n$，选择$(z, r, u, v) \xleftarrow{R} G$并计算

[0137] $$c_1 = e(g_z, z) \cdot e(g_r, r) \cdot \prod_{i=1}^{n} e(g_i, M_i)$$

[0138] $$c_2 = e(h_z, z) \cdot e(h, u) \cdot e(H_G(tag), v) \cdot \prod_{i=1}^{n} e(h_i, M_i)$$。

[0139] 然后返回$com = (c_1, c_2) \in G_T^2$以及 $dec = (z, r, u, v) \in G^4$。

[0140] SSTC.FakeCom(pk，tk，tag)：选择$\hat{M}_1,...,\hat{M}_n \xleftarrow{R} G^n$以及$z,r,u,v \xleftarrow{R} \mathbb{G}$并计算

[0141] $c_1 = e(g_z,z)\cdot e(g_r,r)\cdot\prod_{i=1}^{n} e(g_i,\hat{M}_i)$

[0142] $c_2 = e(h_z,z)\cdot e(h,u)\cdot e(H_G(tag),v)\cdot\prod_{i=1}^{n} e(h_i,\hat{M}_i)$。

[0143] 然后返回$com=(c_1,c_2)\in G_T^2$以及$aux=((\hat{M}_1,...,\hat{M}_n),z,r,u,v)$。

[0144] $\mathrm{SSTC.FakeOpen}\ (aux,tk,tag,\tilde{com},(M_1,...,M_n))$：使用下面的下文中描述的线性同态 SPS 在矢量$(M_1/\hat{M}_1,...,M_n/\hat{M}_n)$上生成线性同态签名（z′，r′，u′ v′）。然后计算$(\tilde{z},\tilde{r},\tilde{u},\tilde{r})=(z\cdot z',r\cdot r',u\cdot u',v\cdot v')$。返回$dec=(\tilde{z},\tilde{r},\tilde{u},\tilde{r})$，其满足

[0145] $c_1 = e(g_z,z)\cdot e(g_r,\tilde{r})\cdot\prod_{i=1}^{n} e(g_i,M_i)$

[0146] $c_2 = e(h_z,\tilde{z})\cdot e(h,\tilde{u})\cdot e(H_G(tag),\tilde{v})\cdot\prod_{i=1}^{n} e(h_i,M_i)$。

[0147] SSTC.Verify(pk，tag，(M_1，...，M_n)，com，dec)：解析该承诺 com 为$(c_1,c_2)\in G_T^2$以及打开 dec 为 (z，r，u，v) ∈ G^4。如果这些未被正确地解析，返回 0。否则，如果 (z，r，u，v)

[0148] $c_1 = e(g_z,z)\cdot e(g_r,r)\cdot\prod_{i=1}^{n} e(g_i,M_i)$

[0149] $c_2 = e(h_z,z)\cdot e(h,u)\cdot e(H_G(tag),v)\cdot\prod_{i=1}^{n} e(h_i,M_i)$

[0150] 返回 1，否则返回 0。

[0151] 本领域技术人员可以理解适用于矢量的本发明的方案，也适用于标量（其维数 n 等于 1)，因此，该维数 n 可以是任何正整数：1，2，3...

[0152] 将理解的是本发明的方案可以提供一个具有所需特征的承诺方案。

[0153] 可以独立地或以任何适当的组合提供说明书和（适当的）权利要求和附图中所公开的每个特征。被描述为以硬件实现的特征也可以被实现为软件，反之亦然。权利要求中出现的附图标记仅仅是举例说明的方式，并不应对权利要求的范围起到任何限制作用。

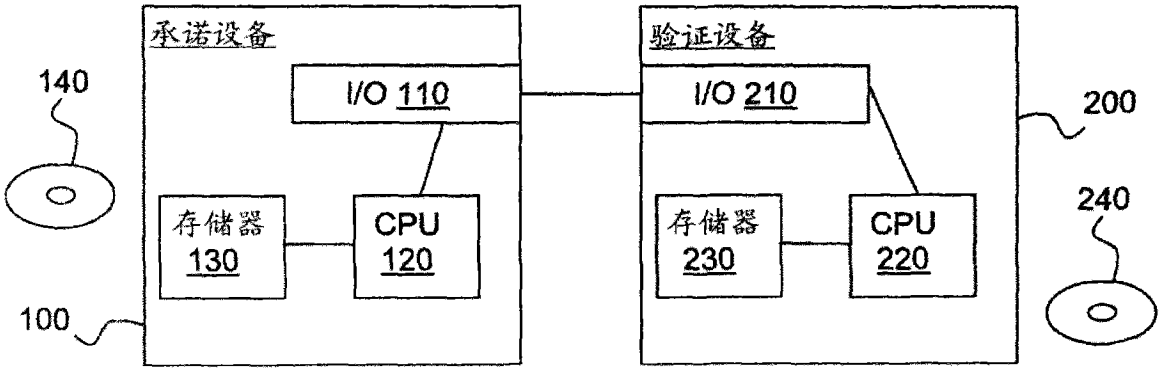

承诺设备
I/O 110
存储器
130
CPU
120
140
100
验证设备
I/O 210
存储器
230
CPU
220
200
240

图 1

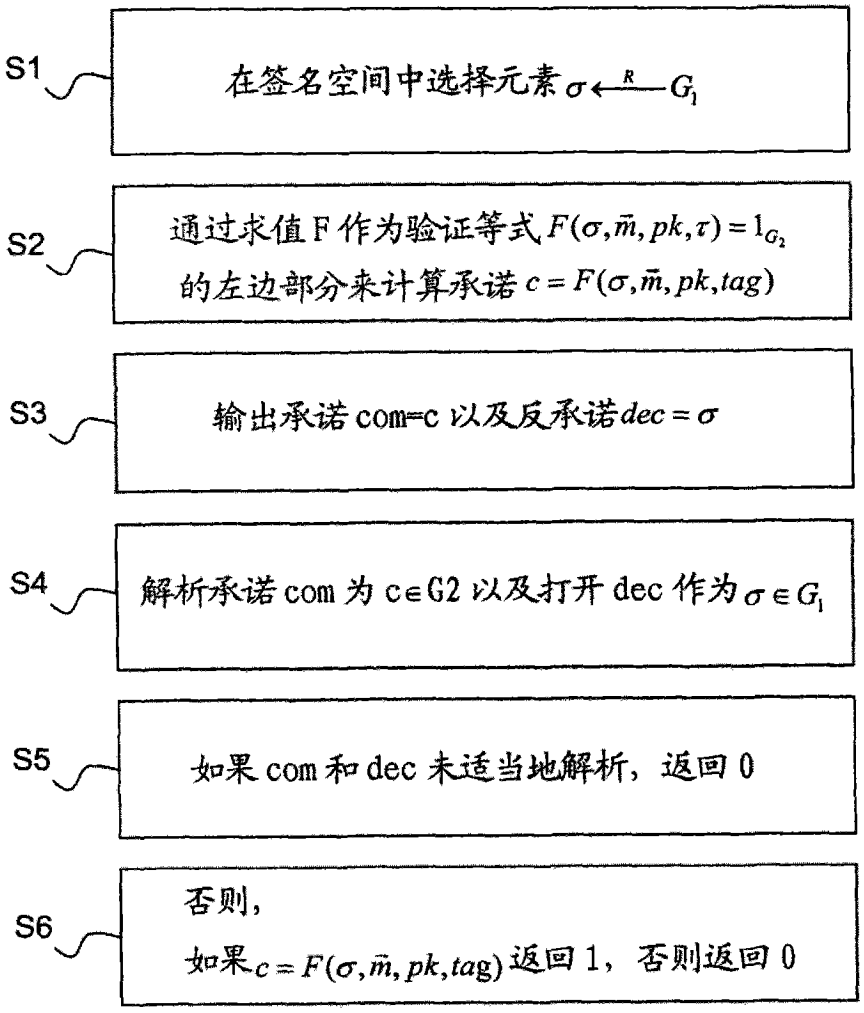

S1
在签名空间中选择元素 $\sigma \xleftarrow{R} G_1$
S2
通过求值 F 作为验证等式 $F(\sigma,\vec{m},pk,\tau)=1_{G_2}$
的左边部分来计算承诺 $c=F(\sigma,\vec{m},pk,tag)$
S3
输出承诺 com=c 以及反承诺 $dec=\sigma$
S4
解析承诺 com 为 c∈G2 以及打开 dec 作为 $\sigma \in G_1$
S5
如果 com 和 dec 未适当地解析，返回 0
S6
否则，
如果 $c=F(\sigma,\vec{m},pk,tag)$ 返回 1，否则返回 0

图 2