



(12) 发明专利

(10) 授权公告号 CN 1633774 B

(45) 授权公告日 2011. 12. 07

(21) 申请号 03803910.9

(22) 申请日 2003. 03. 18

(30) 优先权数据

60/366, 196 2002. 03. 21 US

60/366, 292 2002. 03. 21 US

10/384, 328 2003. 03. 07 US

(85) PCT申请进入国家阶段日

2004. 08. 13

(86) PCT申请的申请数据

PCT/US2003/008010 2003. 03. 18

(87) PCT申请的公布数据

W02003/081780 EN 2003. 10. 02

(73) 专利权人 株式会社 NTT 都科摩

地址 日本东京

(72) 发明人 克雷格·B·金特里

艾丽斯·西尔弗伯格

(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 马浩

(51) Int. Cl.

H04L 9/00 (2006. 01)

H04K 1/00 (2006. 01)

(56) 对比文件

EP 1051036 A2, 2000. 11. 08, 全文.

US 4309569 A, 全文.

CN 1337649 A, 2002. 02. 27, 全文.

US 5590197 A, 1996. 12. 31, 全文.

审查员 李燕

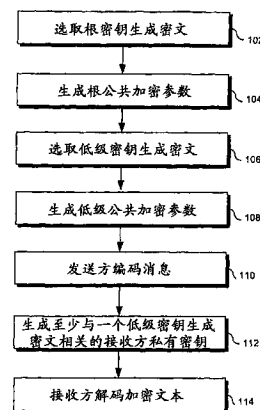
权利要求书 13 页 说明书 17 页 附图 9 页

(54) 发明名称

基于身份的分级加密与签名方案

(57) 摘要

本发明提供了用来在一个包含多个私有密钥生成器 (“PKG”) 的系统中的发送方与接收方之间编码与解码一条数字消息的方法。所述的 PKG 中至少包括一个根 PKG 以及根 PKG 与接收方之间的分级结构中的 n 个低级 PKG。一个根密钥生成密文被选取且只为根 PKG 所知 (102)。根据所述的根密钥生成密文产生一个根密钥生成参数 (104)。为所述的 n 个低级 PKG 各选取一个低级密钥生成密文, 其中每个低级密钥生成密文只对其相关的低级 PKG 已知 (106)。还要为 n 个低级 PKG 各产生一个低级密钥生成参数, 产生的过程中至少要利用对应于其相关低级私有密钥生成器的低级密钥生成密文 (108)。至少利用根密钥生成参数以及接收方相关的接收方身份信息对所述的消息进行编码以形成一个密文 (110)。生成一个接收方私有密钥, 使得该接收方私有密钥至少与根密钥生成密文、n 个低级密钥生成密文中的一个或多个、以及接收方身份信息相关 (112)。至少利用接收方私有密钥对所述的密文解码以恢复所述的消息 (114)。



1. 一种在发送方与接收方之间编码和解码数字消息 M 的方法,其中所述的接收方在一个分级系统中比一个根 PKG 低 $n+1$ 个级别,所述的系统中包括多个 PKG,这多个 PKG 中至少包括一个根 PKG 以及在所述根 PKG 与接收方之间的分级结构中的 n 个低级 PKG,其中 $n \geq 1$,该方法包括:

选取一个仅为根 PKG 所知的根密钥生成密文;

根据所述的根密钥生成密文产生一个根密钥生成参数;

为 n 个低级 PKG 各选取一个低级密钥生成密文,其中每个低级密钥生成密文仅对其相关的低级 PKG 已知;

为 n 个低级 PKG 各产生一个低级密钥生成参数,其中每个低级密钥生成参数的产生至少要利用其相关低级 PKG 的低级密钥生成密文;

至少利用根密钥生成参数和接收方身份信息来对所述的消息进行编码以形成加密文本 C ;

生成一个接收方私有密钥,使得该接收方私有密钥至少与根密钥生成密文、根 PKG 和接收方之间的分级结构中的 n 个低级 PKG 相关的 n 个低级密钥生成密文中的一个或多个、以及接收方身份信息有关;以及

至少利用接收方私有密钥来解密所述的加密文本以恢复所述的消息。

2. 根据权利要求 1 所述的在发送方与接收方之间编码和解码数字消息的方法,其中所述的接收方身份信息由一个接收方 ID 元组 $(ID_{z1}, \dots, ID_{z(n+1)})$ 构成,其中包括与接收方相关的身份信息 $ID_{z(n+1)}$ 以及与根 PKG 和接收方之间的分级结构中的 n 个低级 PKG 相关的身份信息 ID_{zi} 。

3. 根据权利要求 1 所述的在发送方与接收方之间编码和解码数字消息的方法,其中所述的接收方私有密钥与所有的 n 个低级密钥生成密文有关,所述的低级密钥生成密文与根 PKG 和接收方之间的分级结构中的 n 个低级 PKG 相关。

4. 根据权利要求 1 所述的在发送方与接收方之间编码和解码数字消息的方法,其中所述的消息编码还包括保护加密文本免受选定加密文本攻击。

5. 根据权利要求 4 所述的在发送方与接收方之间编码和解码数字消息的方法,其中所述的消息编码还包括:

根据一种对称加密方案加密消息,以形成一个对称加密;以及

根据一种单向加密方案加密所述对称加密,以形成加密文本;并且

其中所述的加密文本解密还包括:

根据所述单向加密方案解密加密文本以恢复所述对称加密;以及

根据所述对称加密方案解密所述对称加密以恢复消息。

6. 权利要求 1 的方法,所述的接收方是与一个接收方 ID 元组 $(ID_{z1}, \dots, ID_{z(n+1)})$ 相关联的实体 z ,该 ID 元组中包含与接收方相关联的身份信息 $ID_{z(n+1)}$ 以及与根 PKG 和接收方之间的分级结构中的 n 个低级 PKG 各自相关的身份信息 ID_{zi} ,该方法还包括:

生成元素的第一循环群 Γ_1 和元素的第二循环群 Γ_2 ;

选取一个函数 $\hat{e}$,该函数能够由第一循环群 Γ_1 的两个元素生成第二循环群 Γ_2 的一个元素;

选取第一循环群 Γ_1 的一个根生成器 P_0 ;

选取一个随机根密钥生成密文 s_0 该密文与根 PKG 相关且只为根 PKG 所知；

产生一个根密钥生成参数 $Q_0 = s_0 P_0$ ；

选取一个第一函数 H_1 ，该函数能够由第一串二进制数生成第一循环群 Γ_1 的一个元素；

选取一个第二函数 H_2 ，该函数能够由第二循环群 Γ_2 的一个元素生成第二串二进制数；

为 n 个低级 PKG 各生成一个公共元素 P_{zi} ，其中 $P_{zi} = H_1(ID_{z1}, \dots, ID_{zi})$ ， $1 \leq i \leq n$ ；

为 n 个低级 PKG 各选取一个低级密钥生成密文 s_{zi} ，其中各个低级密钥生成密文 s_{zi} 只对其相关的低级 PKG 已知；

为 n 个低级 PKG 各生成一个低级机密元素 S_{zi} ，其中 $S_{zi} = S_{z(i-1)} + s_{z(i-1)} P_{zi}$ ， $1 \leq i \leq n$ ，其中 $s_{z0} = s_0$ ，并且其中 S_{z0} 被定义为 0；

为 n 个低级 PKG 各产生一个低级密钥生成参数 Q_{zi} ，其中 $Q_{zi} = s_{zi} P_0$ ， $1 \leq i \leq n$ ；

生成一个与接收方相关的接收方公共元素 $P_{z(n+1)} = H_1(ID_{z1}, \dots, ID_{z(n+1)})$ ；

生成一个与接收方相关的接收方机密元素

$$S_{z(n+1)} = S_{zn} + s_{zn} P_{z(n+1)} = \sum_{i=1}^{n+1} s_{z(i-1)} P_{zi}；$$

至少利用接收方 ID 元组 $(ID_{z1}, \dots, ID_{zi})$ 和根密钥生成参数 Q_0 对消息 M 编码以生成一个加密文本 C ；以及

至少利用接收方机密元素 $S_{z(n+1)}$ 来解码加密文本 C 以恢复消息 M 。

7. 根据权利要求 6 所述的方法，其中：

第一群 Γ_1 和第二群 Γ_2 具有相同的素数阶 q 。

8. 根据权利要求 6 所述的方法，其中：

所述的第一循环群 Γ_1 是超奇异椭圆曲线或阿贝尔簇曲线上的点的加性群，第二循环群 Γ_2 则是一个有限域的乘性子群。

9. 根据权利要求 6 所述的方法，其中：

函数 $\hat{e}$ 是一种双线性、非退化且可高效计算的配对。

10. 根据权利要求 7 所述的方法，其中：

s_0 是循环群 Z/qZ 的一个元素；

Q_0 是第一循环群 Γ_1 的一个元素；

每个公共元素 P_{zi} 都是第一循环群 Γ_1 的一个元素；

每个低级密钥生成密文 s_{zi} 都是循环群 Z/qZ 的一个元素；

每个机密元素 S_{zi} 都是第一循环群 Γ_1 的一个元素；

每个低级密钥生成参数 Q_{zi} 都是第一循环群 Γ_1 的一个元素；

接收方公共元素 $P_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素；并且

接收方机密元素 $S_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素。

11. 根据权利要求 6 所述的方法，其中：编码消息 M 还包括：

选取一个随机参数 r ；并

生成加密文本 $C = [U_0, U_2, \dots, U_{n+1}, V]$ ，其中对于 $i = 0$ 和 $2 \leq i \leq n+1$ 有 $U_i = rP_{zi}$ ，

其中 $V = M \oplus H_2(g^r)$ ，并且其中 $g = \hat{e}(Q_0, P_{z1})$ ；并且

解码加密文本 C 还包括：

利用 $M = V \oplus H_2 \left(\frac{\hat{e}(U_0, S_{z(n+1)})}{\prod_{i=2}^{n+1} \hat{e}(Q_{i-1}, U_i)} \right)$ 恢复消息 M。

12. 根据权利要求 11 所述的方法, 其中:

第一群 Γ_1 和第二群 Γ_2 具有相同的素数阶 q 。

13. 根据权利要求 11 所述的方法, 其中:

所述的第一循环群 Γ_1 是超奇异椭圆曲线或阿贝尔簇曲线上的点的加性群, 第二循环群 Γ_2 则是一个有限域的乘性子群。

14. 根据权利要求 11 所述的方法, 其中:

函数 $\hat{e}$ 是一种双线性、非退化且可高效计算的配对。

15. 根据权利要求 12 所述的方法, 其中:

s_0 是循环群 Z/qZ 的一个元素;

Q_0 是第一循环群 Γ_1 的一个元素;

每个公共元素 P_{zi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成密文 s_{zi} 都是循环群 Z/qZ 的一个元素;

每个机密元素 S_{zi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成参数 Q_{zi} 都是第一循环群 Γ_1 的一个元素;

接收方公共元素 $P_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素;

接收方机密元素 $S_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素;

r 是循环群 Z/qZ 的一个元素; 并且

g 是第二循环群 Γ_2 的一个元素。

16. 根据权利要求 7 所述的方法, 还包括:

选取一个第三函数 H_3 , 该函数能够由第三串二进制数生成循环群 Z/qZ 的一个整数; 以及

选取一个第四函数 H_4 , 该函数能够由第五串二进制数生成第四串二进制数;

其中编码消息 M 还包括:

选取一个随机二进制串 σ ,

选取一个对称加密方案 E;

生成一个随机整数 $r = H_3(\sigma, M, W)$, 其中 $W = E_{H_4(\sigma)}(M)$; 以及

生成加密文本 $C = [U_0, U_2, \dots, U_{n+1}, V, W]$, 其中对于 $i = 0$ 和 $2 \leq i \leq n+1$ 有 $U_i = rP_{zi}$,

其中 $V = \sigma \oplus H_2(g')$, 并且其中 $g = \hat{e}(Q_0, P_{zi})$; 并且

其中解码加密文本 C 还包括:

利用 $\sigma = V \oplus H_2 \left(\frac{\hat{e}(U_0, S_{z(n+1)})}{\prod_{i=2}^{n+1} \hat{e}(Q_{i-1}, U_i)} \right)$ 恢复随机二进制串 σ , 并

利用 $M = E_{H_4(\sigma)}^{-1}(W)$ 恢复消息 M。

17. 根据权利要求 16 所述的方法, 其中:

第一循环群 Γ_1 和第二循环群 Γ_2 具有相同的素数阶 q 。

18. 根据权利要求 16 所述的方法, 其中:

所述的第一循环群 Γ_1 是超奇异椭圆曲线或阿贝尔簇曲线上的点的加性群,第二循环群 Γ_2 则是一个有限域的乘性子群。

19. 根据权利要求 16 所述的方法,其中:

函数 $\hat{e}$ 是一种双线性、非退化且可高效计算的配对。

20. 根据权利要求 16 所述的方法,还包括:

通过计算一个实验性的随机整数 $r' = H_3(\sigma, M, W)$; 以及

对 $2 \leq i \leq n+1$ 检验 $U_0 = r' \cdot P_0$ 和 $U_i = r' \cdot P_{zi}$ 来检验加密文本 C 的内部一致性。

21. 根据权利要求 17 所述的方法,其中:

s_0 是循环群 Z/qZ 的一个元素;

Q_0 是第一循环群 Γ_1 的一个元素;

每个公共元素 P_{zi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成密文 s_{zi} 都是循环群 Z/qZ 的一个元素;

每个机密元素 S_{zi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成参数 Q_{zi} 都是第一循环群 Γ_1 的一个元素;

接收方公共元素 $P_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素;

接收方机密元素 $S_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素;

r 是循环群 Z/qZ 的一个元素; 并且

g 是第二循环群 Γ_2 的一个元素。

22. 一种在一个系统中的发送方与接收方之间编码和解码消息 M 的方法,所述的系统中包括多个 PKG,这多个 PKG 中包括根 PKG 与发送方之间的分级结构中的 m 个低级 PKG,其中 $m \geq 1$,以及根 PKG 与接收方之间的分级结构中的 n 个低级 PKG,其中 $n \geq 1$,所述分级结构中的多个 PKG 中至少有 1 个是发送方与接收方共同的前辈 PKG,其中 $1 \geq 1$,其中的 PKG_1 是发送方与接收方共同的前辈 PKG,该方法还包括:

选取一个仅为根 PKG 所知的根密钥生成密文;

根据所述的根密钥生成密文产生一个根密钥生成参数;

为 m 和 n 个低级 PKG 各选取一个低级密钥生成密文,其中每个低级密钥生成密文仅对其相关的低级 PKG 已知;

为 m 和 n 个低级 PKG 各产生一个低级密钥生成参数,其中每个低级密钥生成参数的产生至少要利用其相关低级 PKG 的低级密钥生成密文;

生成一个发送方私有密钥,使得该发送方私有密钥至少与发送方身份信息、根密钥生成密文、以及在根 PKG 和发送方之间的分级结构中的 m 个低级 PKG 相关的 m 个低级密钥生成密文中的一个或多个有关;

生成一个接收方私有密钥,使得该接收方私有密钥至少与接收方身份信息、根密钥生成密文、以及在根 PKG 和接收方之间的分级结构中的 n 个低级 PKG 相关的 n 个低级密钥生成密文中的一个或多个有关;

至少利用接收方身份信息、发送方私有密钥、以及和位于或低于公共前辈 PKG_1 级别的 $(m-1+1)$ 个 PKG 相关的低级密钥生成参数中的 0 个或多个,来对所述的消息进行编码以形成加密文本 C,但是不使用和高于公共前辈 PKG_1 的 $(1-1)$ 个 PKG 相关的低级密钥生成参数中的任何一个; 以及

至少利用接收方私有密钥以及和位于或低于最低公共前辈 PKG_1 级别的 $(n-1+1)$ 个 PKG 相关的低级密钥生成参数中的 0 个或多个,来对所述的消息进行解码,但是不能使用和高于公共前辈 PKG_1 的 $(1-1)$ 个 PKG 相关的低级密钥生成参数中的任何一个。

23. 根据权利要求 22 所述的在一个系统中的发送方与接收方之间编码和解码消息的方法:

其中所述的消息编码还包括利用与根 PKG 和发送方之间位于或低于公共前辈 PKG_1 级别的 $(m-1+1)$ 个 PKG 相关的低级密钥生成参数中的一个或多个;并且

其中所述的消息解码还包括利用与根 PKG 和接收方之间位于或低于最低公共前辈 PKG_1 级别的 $(n-1+1)$ 个 PKG 相关的低级密钥生成参数中的一个或多个。

24. 根据权利要求 22 所述的在一个系统中的发送方与接收方之间编码和解码消息的方法:

其中所述的消息编码还包括利用与根 PKG 和发送方之间位于或低于公共前辈 PKG_1 级别的 $(m-1+1)$ 个 PKG 相关的低级密钥生成参数中的一个或多个;并且

其中所述的消息解码还包括利用与根 PKG 和接收方之间位于或低于最低公共前辈 PKG_1 级别的 $(n-1+1)$ 个 PKG 相关的低级密钥生成参数中的零个。

25. 根据权利要求 22 所述的在一个系统中的发送方与接收方之间编码和解码消息的方法:

其中所述的消息编码还包括利用与根 PKG 和发送方之间位于或低于公共前辈 PKG_1 级别的 $(m-1+1)$ 个 PKG 相关的低级密钥生成参数中的零个;并且

其中所述的消息解码还包括利用与根 PKG 和接收方之间位于或低于最低公共前辈 PKG_1 级别的 $(n-1+1)$ 个 PKG 相关的低级密钥生成参数中的一个或多个。

26. 根据权利要求 22 所述的在一个系统中的发送方与接收方之间编码和解码消息的方法,其中要利用与根 PKG 和发送方之间位于或低于公共前辈 PKG_1 级别的 $(m-1+1)$ 个 PKG 相关的所有低级密钥生成参数来对消息进行编码。

27. 根据权利要求 22 所述的在一个系统中的发送方与接收方之间编码和解码消息的方法,其中要利用与根 PKG 和接收方之间位于或低于公共前辈 PKG_1 级别的 $(n-1+1)$ 个 PKG 相关的所有低级密钥生成参数来对消息进行解码。

28. 权利要求 22 的方法,其中发送方是与一个发送方 ID 元组 $(\text{ID}_{y1}, \dots, \text{ID}_{y(m+1)})$ 相关联的实体 y ,该发送方 ID 元组中包含与发送方 y 有关的信息 $\text{ID}_{y(m+1)}$ 以及与根 PKG 和发送方 y 之间的分级结构中的 m 个低级 PKG 相关的身份信息 ID_{yi} ,并且其中接收方 z 与一个接收方 ID 元组 $(\text{ID}_{z1}, \dots, \text{ID}_{z(n+1)})$ 相关联,该接收方 ID 元组中包含与接收方有关的身份信息 $\text{ID}_{z(n+1)}$ 以及与根 PKG 和接收方之间的分级结构中的 n 个低级 PKG 相关的身份信息 ID_{zi} ,该方法还包括:

生成元素的第一循环群 Γ_1 和元素的第二循环群 Γ_2 ;

选取一个函数 $\hat{e}$,该函数能够由第一循环群 Γ_1 的两个元素生成第二循环群 Γ_2 的一个元素;

选取第一循环群 Γ_1 的一个根生成器 P_0 ;

选取一个随机根密钥生成密文 s_0 ,该密文与根 PKG 相关且只为根 PKG 所知;

产生一个根密钥生成参数 $Q_0 = s_0 P_0$;

选取一个第一函数 H_1 , 该函数能够由第一串二进制数生成第一循环群 Γ_1 的一个元素;
 选取一个第二函数 H_2 , 该函数能够由第二循环群 Γ_2 的一个元素生成第二串二进制数;
 为 m 个低级 PKG 各生成一个公共元素 P_{yi} , 其中 $P_{yi} = H_1(ID_{y1}, \dots, ID_{yi}), 1 \leq i \leq m$, 并且对于所有的 $i \leq 1$ 有 $P_{yi} = P_{zi}$;

为 n 个低级 PKG 各生成一个公共元素 P_{zi} , 其中 $P_{zi} = H_1(ID_{z1}, \dots, ID_{zi}), 1 \leq i \leq n$;

为 m 个低级 PKG 各选取一个低级密钥生成密文 s_{yi} , 其中对于所有的 $i \leq 1$ 有 $s_{yi} = s_{zi}$;

为 n 个低级 PKG 各选取一个低级密钥生成密文 s_{zi} , 其中各个低级密钥生成密文 s_{zi} 只对其相关的低级 PKG 已知;

为 m 个低级 PKG 各生成一个低级机密元素 S_{yi} , 其中 $S_{yi} = S_{y(i-1)} + s_{y(i-1)} P_{yi}, 1 \leq i \leq m$, 并且对于所有的 $i \leq 1$ 有 $S_{yi} = S_{zi}$;

为 n 个低级 PKG 各生成一个低级机密元素 S_{zi} , 其中 $S_{zi} = S_{z(i-1)} + s_{z(i-1)} P_{zi}, 1 \leq i \leq n$, 其中 $s_{z0} = s_0$, 并且其中 S_{z0} 被定义为 0;

为 m 个低级 PKG 各产生一个低级密钥生成参数 Q_{yi} , 其中 $Q_{yi} = s_{yi} P_0, 1 \leq i \leq m$, 并且其中对于所有的 $i \leq 1$ 有 $Q_{yi} = Q_{zi}$;

为 n 个低级 PKG 各产生一个低级密钥生成参数 Q_{zi} , 其中 $Q_{zi} = s_{zi} P_0, 1 \leq i \leq n$;

生成一个与发送方 y 相关的发送方公共元素 $P_{y(m+1)} = H_1(ID_{y1}, \dots, ID_{y(m+1)})$;

生成一个与接收方相关的接收方公共元素 $P_{z(n+1)} = H_1(ID_{z1}, \dots, ID_{z(n+1)})$;

生成一个与发送方相关的发送方机密元素

$$S_{y(m+1)} = S_{ym} + s_{ym} P_{y(m+1)} = \sum_{i=1}^{m+1} s_{y(i-1)} P_{yi};$$

生成一个与接收方相关的接收方机密元素

$$S_{z(n+1)} = S_{zn} + s_{zn} P_{z(n+1)} = \sum_{i=1}^{n+1} s_{z(i-1)} P_{zi};$$

至少利用对应于 $1 < i \leq m$ 的低级密钥生成参数 Q_{yi} 和发送方机密元素 $S_{y(m+1)}$ 对消息 M 编码以生成一个加密文本 C , 但不能使用对应于 $i < 1$ 的低级密钥生成参数 Q_{yi} ; 以及

至少利用对应于 $1 < i \leq n$ 的低级密钥生成参数 Q_{zi} 和接收方机密元素 $S_{z(n+1)}$ 对加密文本 C 解码以恢复消息 M , 但不能使用对应于 $i < 1$ 的低级密钥生成参数 Q_{zi} 。

29. 根据权利要求 28 所述的方法, 其中编码消息 M 还包括使用低级密钥生成参数 Q_{y1} 。

30. 根据权利要求 28 所述的方法, 其中解码消息 M 还包括使用低级密钥生成参数 Q_{z1} 。

31. 根据权利要求 28 所述的方法, 其中:

第一群 Γ_1 和第二群 Γ_2 具有相同的素数阶 q 。

32. 根据权利要求 28 所述的方法, 其中:

所述的第一循环群 Γ_1 是超奇异椭圆曲线或阿贝尔簇曲线上的点的加性群, 第二循环群 Γ_2 则是一个有限域的乘性子群。

33. 根据权利要求 28 所述的方法, 其中:

函数 $\hat{e}$ 是一种双线性、非退化且可高效计算的配对。

34. 根据权利要求 31 所述的方法, 其中:

s_0 是循环群 Z/qZ 的一个元素;

Q_0 是第一循环群 Γ_1 的一个元素;

每个公共元素 P_{zi} 都是第一循环群 Γ_1 的一个元素;

每个公共元素 P_{yi} 都是第一循环群 Γ_1 的一个元素；
 每个低级密钥生成密文 s_{zi} 都是循环群 Z/qZ 的一个元素；
 每个低级密钥生成密文 s_{yi} 都是循环群 Z/qZ 的一个元素；
 每个机密元素 S_{zi} 都是第一循环群 Γ_1 的一个元素；
 每个机密元素 S_{yi} 都是第一循环群 Γ_1 的一个元素；
 每个低级密钥生成参数 Q_{zi} 都是第一循环群 Γ_1 的一个元素；
 每个低级密钥生成参数 Q_{yi} 都是第一循环群 Γ_1 的一个元素；
 接收方公共元素 $P_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素；
 发送方公共元素 $P_{y(m+1)}$ 是第一循环群 Γ_1 的一个元素；
 接收方机密元素 $S_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素；
 发送方机密元素 $S_{y(m+1)}$ 是第一循环群 Γ_1 的一个元素。

35. 根据权利要求 28 所述的方法，

其中编码消息 M 还包括：

选取一个随机参数 r ；并

编码消息 M 以生成加密文本 $C = [U_0, U_2, \dots, U_{n+1}, V]$ ，其中 $U_0 = rP_0$ ，其中对于

$1+1 \leq i \leq n+1$ 有 $U_i = rP_{zi}$ ，其中 $V = M \oplus H_2(g_{yl}^r)$ ，并且其中 $g_{yl} = \frac{\hat{e}(P_0, S_{y(m+1)})}{\prod_{i=l+1}^{m+1} \hat{e}(Q_{y(i-1)}, P_{yi})}$ ；并且

解码加密文本 C 还包括：

利用 $M = V \oplus H_2\left(\frac{\hat{e}(U_0, S_{z(n+1)})}{\prod_{i=l+1}^{n+1} \hat{e}(Q_{z(i-1)}, U_{zi})}\right)$ 恢复消息 M 。

36. 根据权利要求 35 所述的方法，其中：

第一循环群 Γ_1 和第二循环群 Γ_2 具有相同的素数阶 q 。

37. 根据权利要求 35 所述的方法，其中：

所述的第一循环群 Γ_1 是超奇异椭圆曲线或阿贝尔簇曲线上的点的加性群，第二循环群 Γ_2 则是一个有限域的乘性子群。

38. 根据权利要求 35 所述的方法，其中：

函数 $\hat{e}$ 是一种双线性、非退化且可高效计算的配对。

39. 根据权利要求 36 所述的方法，其中：

s_0 是循环群 Z/qZ 的一个元素；

Q_0 是第一循环群 Γ_1 的一个元素；

每个公共元素 P_{zi} 都是第一循环群 Γ_1 的一个元素；

每个公共元素 P_{yi} 都是第一循环群 Γ_1 的一个元素；

每个低级密钥生成密文 s_{zi} 都是循环群 Z/qZ 的一个元素；

每个低级密钥生成密文 s_{yi} 都是循环群 Z/qZ 的一个元素；

每个机密元素 S_{zi} 都是第一循环群 Γ_1 的一个元素；

每个机密元素 S_{yi} 都是第一循环群 Γ_1 的一个元素；

每个低级密钥生成参数 Q_{zi} 都是第一循环群 Γ_1 的一个元素；

每个低级密钥生成参数 Q_{yi} 都是第一循环群 Γ_1 的一个元素；

接收方公共元素 $P_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素；
 发送方公共元素 $P_{y(m+1)}$ 是第一循环群 Γ_1 的一个元素；
 接收方机密元素 $S_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素；
 发送方机密元素 $S_{y(m+1)}$ 是第一循环群 Γ_1 的一个元素；
 r 是循环群 Z/qZ 的一个元素；并且

g_{y1} 是第二循环群 Γ_2 的一个元素。

40. 根据权利要求 28 所述的方法：

其中编码消息 M 还包括：

选取一个随机参数 r ；并

编码消息 M 以生成加密文本 $C = [U_0, U_2, \dots, U_{n+1}, V]$ ，其中 $U_0 = rP_0$ ，其中 $U_{l+1} = r(P_{y(l+1)} - P_{z(l+1)})$ ，其中对于 $l+2 \leq i \leq n$ 有 $U_i = rP_{zi}$ ，其中 $V = M \oplus H_2(g'_{y(l+1)})$ ，并且其中

$$g_{y(l+1)} = \frac{\hat{e}(P_0, S_{y(m+1)})}{\prod_{i=l+2}^m \hat{e}(Q_{y(i-1)}, P_{yi})} = \hat{e}(P_0, S_{y(l+1)}); \text{ 并且}$$

解码加密文本 C 还包括：

$$\text{利用 } M = V \oplus H_2 \left(\frac{\hat{e}(U_0, S_{z(n+1)}) \hat{e}(U_{l+1}, Q_{zl})}{\prod_{i=l+2}^n \hat{e}(Q_{z(i-1)}, U_i)} \right) \text{ 恢复消息 } M。$$

41. 根据权利要求 28 所述的方法：

其中编码消息 M 还包括：

选取一个随机参数 r ；并

编码消息 M 以生成加密文本 $C = [U_0, U_{l+2}, \dots, U_n, V]$ ，其中 $U_0 = rP_0$ ，其中对于 $l+2 \leq i \leq n$ 有 $U_i = rP_{zi}$ ，其中 $V = M \oplus H_2(g'_{z(l+1)})$ ，并且其中

$$g_{z(l+1)} = \frac{\hat{e}(P_0, S_{y(m+1)}) \hat{e}(Q_{yl}, (P_{z(l+1)} - P_{y(l+1)}))}{\prod_{i=l+2}^m \hat{e}(Q_{y(i-1)}, P_{yi})} = \hat{e}(P_0, S_{z(l+1)}); \text{ 并且}$$

解码加密文本 C 还包括：

$$\text{利用 } M = V \oplus H_2 \left(\frac{\hat{e}(U_0, S_{z(n+1)})}{\prod_{i=l+2}^n \hat{e}(Q_{z(i-1)}, U_i)} \right) \text{ 恢复消息 } M。$$

42. 根据权利要求 31 所述的方法，还包括：

选取一个第三函数 H_3 ，该函数能够由第三串二进制数生成循环群 Z/qZ 的一个整数；以及

选取一个第四函数 H_4 ，该函数能够由第五串二进制数生成第四串二进制数；

其中编码消息 M 还包括：

选取一个随机二进制串即 σ ，

计算一个随机整数 $r = H_3(\sigma, M, W)$ ，其中 $W = E_{H_4(\sigma)}(M)$ ；以及

生成加密文本 $C = [U_0, U_{l+1}, \dots, U_{n+1}, V, W]$ ，其中对于 $i = 0$ 和 $l+1 \leq i \leq n+1$ 有 $U_i =$

rP_{zi} , 其中 $V = \sigma \oplus H_2(g_{yl}^r)$, 并且其中 $g_{yl} = \frac{\hat{e}(P_0, S_{y(m+1)})}{\prod_{i=l+1}^{m+1} \hat{e}(Q_{y(i-1)}, P_{yi})}$; 并且

其中解码加密文本 C 还包括:

利用 $\sigma = V \oplus H_2\left(\frac{\hat{e}(U_0, S_{z(n+1)})}{\prod_{i=l+1}^{n+1} \hat{e}(Q_{z(i-1)}, U_{zi})}\right)$ 恢复随机二进制串 σ , 并

利用 $M = E_{H_4(\sigma)}^{-1}(W)$ 恢复消息 M。

43. 根据权利要求 41 所述的方法, 其中:

第一循环群 Γ_1 和第二循环群 Γ_2 具有相同的素数阶 q 。

44. 根据权利要求 41 所述的方法, 其中:

所述的第一循环群 Γ_1 是超奇异椭圆曲线或阿贝尔簇曲线上的点的加性群, 第二循环群 Γ_2 则是一个有限域的乘性子群。

45. 根据权利要求 41 所述的方法, 其中:

函数 $\hat{e}$ 是一种双线性、非退化且可高效计算的配对。

46. 根据权利要求 43 所述的方法, 其中:

s_0 是循环群 Z/qZ 的一个元素;

Q_0 是第一循环群 Γ_1 的一个元素;

每个公共元素 P_{zi} 都是第一循环群 Γ_1 的一个元素;

每个公共元素 P_{yi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成密文 s_{zi} 都是循环群 Z/qZ 的一个元素;

每个低级密钥生成密文 s_{yi} 都是循环群 Z/qZ 的一个元素;

每个机密元素 S_{zi} 都是第一循环群 Γ_1 的一个元素;

每个机密元素 S_{yi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成参数 Q_{zi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成参数 Q_{yi} 都是第一循环群 Γ_1 的一个元素;

接收方公共元素 $P_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素;

发送方公共元素 $P_{y(m+1)}$ 是第一循环群 Γ_1 的一个元素;

接收方机密元素 $S_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素;

发送方机密元素 $S_{y(m+1)}$ 是第一循环群 Γ_1 的一个元素;

r 是循环群 Z/qZ 的一个元素; 并且

$g_{z(l+1)}$ 是第二循环群 Γ_2 的一个元素。

47. 根据权利要求 42 所述的方法, 还包括:

通过计算一个实验性的随机整数 $r' = H_3(\sigma, M, W)$; 以及

对 $l+1 \leq i \leq n+1$ 检验 $U_0 = r' \cdot P_0$ 和 $U_i = r' \cdot P_{zi}$ 来检验加密文本 C 的内部一致性。

48. 一种在发送方与接收方之间产生及验证一条消息 M 的数字签名的方法, 其中所述的发送方在分级系统中比根 PKG 低 $m+1$ 个级别, 所述的系统中包括多个 PKG, 这多个 PKG 中至少包括一个根 PKG 以及根 PKG 与发送方之间的分级结构中的 m 个低级 PKG, 其中 $m \geq 1$, 该方法包括:

选取一个仅为根 PKG 所知的根密钥生成密文;

根据所述的根密钥生成密文产生一个根密钥生成参数；

为 m 个低级 PKG 各产生一个低级密钥生成密文，其中每个低级密钥生成密文仅对其相关的低级 PKG 已知；

为 m 个低级 PKG 各产生一个低级密钥生成参数，其中每个低级密钥生成参数的产生至少要利用其相关低级 PKG 的低级密钥生成密文；

为发送方生成一个发送方私有密钥，使得该发送方私有密钥至少与发送方身份信息、根密钥生成密文、以及根 PKG 和发送方之间的分级结构中的 m 个低级 PKG 相关的 m 个低级密钥生成密文中的一个或多个有关；

至少利用发送方私有密钥签署所述的消息以生成数字签名 Sig；以及

至少利用根密钥生成参数和发送方身份信息验证所述的数字签名。

49. 根据权利要求 48 所述的在发送方与接收方之间产生及验证一条消息的数字签名的方法，其中：

还利用一个或多个低级密钥生成参数来验证数字签名。

50. 权利要求 49 的方法，其中所述的发送方与一个发送方 ID 元组 $(ID_{y1}, \dots, ID_{y(m+1)})$ 相关联，该 ID 元组中包含与发送方相关的身份信息 $ID_{y(m+1)}$ ，以及与根 PKG 和发送方之间的分级结构中的 m 个低级 PKG 各自相关的身份信息 ID_{yi} ，该方法还包括：

生成元素的第一循环群 Γ_1 和元素的第二循环群 Γ_2 ；

选取一个双线性、非退化的配对 $\hat{e}$ ，该配对能够由第一循环群 Γ_1 的两个元素生成第二循环群 Γ_2 的一个元素；

选取第一循环群 Γ_1 的一个根生成器 P_0 ；

选取一个随机根密钥生成密文 s_0 ，该密文与根 PKG 相关且只为根 PKG 所知；

产生一个根密钥生成参数 $Q_0 = s_0 P_0$ ；

选取一个第一函数 H_1 ，该函数能够由第一串二进制数生成第一循环群 Γ_1 的一个元素；

为所述的 m 个低级 PKG 各生成一个公共元素 P_{yi} ，其中 $P_{yi} = H_1(ID_{y1}, \dots, ID_{yi})$ ， $1 \leq i \leq m$ ；

为所述的 n 个低级 PKG 各选取一个低级密钥生成密文 s_{yi} ，其中各个低级密钥生成密文 s_{yi} 只对与它相关的低级 PKG 已知；

为所述的 m 个低级 PKG 各生成一个低级机密元素 S_{yi} ，其中 $S_{yi} = S_{y(i-1)} + s_{y(i-1)} P_{yi}$ ， $1 \leq i \leq m$ ；

为所述的 m 个低级 PKG 各产生一个低级密钥生成参数 Q_{yi} ，其中 $Q_{yi} = s_{yi} P_0$ ， $1 \leq i \leq m$ ；

生成一个与发送方相关的发送方公共元素 $P_{y(m+1)} = H_1(ID_{y1}, \dots, ID_{y(m+1)})$ ；

生成一个与发送方相关的发送方机密元素

$$S_{y(m+1)} = S_{ym} + s_{ym} P_{y(m+1)} = \sum_{i=1}^{m+1} s_{y(i-1)} P_{yi} ;$$

至少利用发送方机密元素 $S_{y(m+1)}$ 签署消息 M 以生成一个数字签名 Sig；以及

至少利用根密钥生成参数 Q_0 和低级密钥生成参数 Q_{yi} 验证数字签名 Sig。

51. 根据权利要求 50 所述的方法，其中：

第一群 Γ_1 和第二群 Γ_2 具有相同的素数阶 q 。

52. 根据权利要求 50 所述的方法，其中：

所述的第一循环群 Γ_1 是超奇异椭圆曲线或阿贝尔簇曲线上的点的加性群，第二循环

群 Γ_2 则是一个有限域的乘性子群。

53. 根据权利要求 50 所述的方法, 其中:

函数 $\hat{e}$ 是一种双线性、非退化且可高效计算的配对。

54. 根据权利要求 51 所述的方法, 其中:

s_0 是循环群 Z/qZ 的一个元素;

Q_0 是第一循环群 Γ_1 的一个元素;

每个公共元素 P_{yi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成密文 s_{yi} 都是循环群 Z/qZ 的一个元素;

每个机密元素 S_{yi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成参数 Q_{yi} 都是第一循环群 Γ_1 的一个元素;

发送方公共元素 $P_{y(m+1)}$ 是第一循环群 Γ_1 的一个元素; 并且

发送方机密元素 $S_{y(m+1)}$ 是第一循环群 Γ_1 的一个元素。

55. 根据权利要求 50 所述的方法, 还包括:

选取一个第二函数 H_3 , 该函数能够由第二串二进制数生成第一循环群 Γ_1 的一个元素;

为发送方 y 选取一个发送方密钥生成密文 $s_{y(m+1)}$, 其中所述的发送方密钥生成密文 $s_{y(m+1)}$ 仅对发送方已知; 以及

产生一个与发送方相关的发送方密钥生成参数 $Q_{y(m+1)}$, 其中 $Q_{y(m+1)} = s_{y(m+1)} P_0$;

其中签署消息 M 还包括:

生成一个消息要素 $P_M = H_3(ID_{y1}, \dots, ID_{y(m+1)}, M)$, 其中所述的消息要素 P_M 是第一循环群 Γ_1 的一个元素; 以及

利用 $Sig = S_{y(m+1)} + s_{y(m+1)} P_M$ 生成数字签名 Sig ; 并且

其中验证数字消息 Sig 还包括:

$$\text{检验 } \frac{\hat{e}(P_0, Sig)}{\hat{e}(Q_{y(m+1)}, P_M) \prod_{i=2}^{m+1} \hat{e}(Q_{y(i-1)}, P_{yi})} = \hat{e}(Q_0, P_i)。$$

56. 根据权利要求 55 所述的方法, 其中:

第一群 Γ_1 和第二群 Γ_2 具有相同的素数阶 q 。

57. 根据权利要求 55 所述的方法, 其中:

所述的第一循环群 Γ_1 是超奇异椭圆曲线或阿贝尔簇曲线上的点的加性群, 第二循环群 Γ_2 则是一个有限域的乘性子群。

58. 根据权利要求 55 所述的方法, 其中:

函数 $\hat{e}$ 是一种双线性、非退化且可高效计算的配对。

59. 根据权利要求 56 所述的方法, 其中:

s_0 是循环群 Z/qZ 的一个元素;

Q_0 是第一循环群 Γ_1 的一个元素;

每个公共元素 P_{yi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成密文 s_{yi} 和发送方密钥生成密文 $s_{y(m+1)}$ 都是循环群 Z/qZ 的一个元素;

每个机密元素 S_{yi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成参数 Q_{y_i} 以及发送方密钥生成参数 $Q_{y(m+1)}$ 都是第一循环群 Γ_1 的一个元素；

发送方公共元素 $P_{y(m+1)}$ 是第一循环群 Γ_1 的一个元素；并且

发送方机密元素 $S_{y(m+1)}$ 是第一循环群 Γ_1 的一个元素。

60. 一种为一个系统中的一个实体生成一个私有密钥的方法，所述的系统中包括多个 PKG，这多个 PKG 中至少包括一个根 PKG 以及根 PKG 与所述实体之间的分级结构中的 n 个低级 PKG，其中 $n \geq 1$ ，该方法包括：

生成一个仅为根 PKG 所知的根密钥生成密文；

根据所述的根密钥生成密文产生一个根密钥生成参数；

为 n 个低级 PKG 各产生一个低级密钥生成密文，其中每个低级密钥生成密文仅对其相关的低级 PKG 已知；

为 n 个低级 PKG 各产生一个低级密钥生成参数，其中每个低级密钥生成参数的产生至少要利用其相关低级 PKG 的低级密钥生成密文；

为所述实体生成一个私有密钥，使得该私有密钥至少与和实体相关的身份信息、根密钥生成密文、以及根 PKG 和所述实体之间的分级结构中的 n 个低级 PKG 相关的 n 个低级密钥生成密文中的一个或多个有关；以及

将所述的私有密钥提供给所述的实体。

61. 权利要求 60 的方法，其中所述实体是接收方 z ，所述的接收方 z 在分级结构中比根 PKG 低 $n+1$ 个级别，并且所述的接收方与一个接收方 ID 元组 $(ID_{z1}, \dots, ID_{z(n+1)})$ 相关联，该 ID 元组中包括与接收方相关的身份信息 $ID_{z(n+1)}$ 以及与根 PKG 和接收方之间的分级结构中的 n 个低级 PKG 相关的身份信息 ID_{zi} ，该方法还包括：

生成元素的第一循环群 Γ_1 与元素的第二循环群 Γ_2 ；

选取一个函数 $\hat{e}$ ，该函数能够由第一循环群 Γ_1 的两个元素生成第二循环群 Γ_2 的一个元素；

选取第一循环群 Γ_1 的一个根生成器 P_0 ；

选取一个随机根密钥生成密文 s_0 ，该密文与根 PKG 相关且只为根 PKG 所知；

产生一个根密钥生成参数 $Q_0 = s_0 P_0$ ；

选取一个函数 H_1 ，该函数能够由第一串二进制数生成第一循环群 Γ_1 的一个元素；

为 n 个低级 PKG 各生成一个公共元素 P_{zi} ，其中 $P_{zi} = H_1(ID_{z1}, \dots, ID_{zi})$ ， $1 \leq i \leq n$ ；

为 n 个低级 PKG 各选取一个低级密钥生成密文 s_{zi} ，其中各个低级密钥生成密文 s_{zi} 只对其相关的低级 PKG 已知；

为 n 个低级 PKG 各生成一个低级机密元素 S_{zi} ，其中 $S_{zi} = S_{z(i-1)} + s_{z(i-1)} P_{zi}$ ， $1 \leq i \leq n$ ，其中 $s_{z0} = s_0$ ，并且其中 S_{z0} 被定义为 0；

为 n 个低级 PKG 各产生一个低级密钥生成参数 Q_{zi} ，其中 $Q_{zi} = s_{zi} P_0$ ， $1 \leq i \leq n$ ；

生成与接收方相关的接收方公共元素 $P_{z(n+1)} = H_1(ID_{z1}, \dots, ID_{z(n+1)})$ ，其中 $P_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素；以及

生成与接收方相关的接收方私有密钥

$$S_{z(n+1)} = S_{zn} + s_{zn} P_{z(n+1)} = \sum_{i=1}^{n+1} s_{z(i-1)} P_{zi} \circ$$

62. 根据权利要求 61 所述的方法, 其中:

第一群 Γ_1 和第二群 Γ_2 具有相同的素数阶 q 。

63. 根据权利要求 61 所述的方法, 其中:

所述的第一循环群 Γ_1 是超奇异椭圆曲线或阿贝尔簇曲线上的点的加性群, 第二循环群 Γ_2 则是一个有限域的乘性子群。

64. 根据权利要求 61 所述的方法, 其中:

函数 $\hat{e}$ 是一种可行的配对。

65. 根据权利要求 62 所述的方法, 其中:

s_0 是循环群 Z/qZ 的一个元素;

Q_0 是第一循环群 Γ_1 的一个元素;

每个公共元素 P_{zi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成密文 s_{zi} 都是循环群 Z/qZ 的一个元素;

每个机密元素 S_{zi} 都是第一循环群 Γ_1 的一个元素;

每个低级密钥生成参数 Q_{zi} 都是第一循环群 Γ_1 的一个元素;

接收方公共元素 $P_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素; 并且

接收方私有密钥 $S_{z(n+1)}$ 是第一循环群 Γ_1 的一个元素。

基于身份的分级加密与签名方案

[0001] 相关申请

[0002] 申请人特此依据 35 U.S.C § 119(e) 要求 2002 年 3 月 21 日提交的临时美国专利申请 60/366292 和 2002 年 3 月 21 日提交的临时美国专利申请 60/366196 的优先权,所述的两个临时专利申请都通过引用包括在本申请中。

技术领域

[0003] 本发明主要涉及密码技术以及通过计算机网络或通过其他类型的系统与设备进行的保密通信,并尤其涉及用来对通信进行加密和解密的基于身份的分级方案。

背景技术

[0004] 大致说来,基于身份的密码系统是公共密钥密码系统,在这类系统中,一个实体的公共密钥是由与该实体的身份相关的信息得来的。例如,所述的身份信息可以是个人信息(即姓名、地址、电子邮箱地址等)或是计算机信息(即 IP 地址等)。但是,身份信息不仅可以包括与实体身份严格相关的信息,还包括广泛的可用信息,比如时间或日期。也就是说,身份信息概念的重要性不在于它与实体身份的严格关系,而在于任何希望向实体发送加密消息的人都能轻易获得该信息。

[0005] 一个实体的私有密钥由一个受委托方或逻辑进程产生并分配,所述的受委托方或逻辑进程通常被称为私有密钥生成器(“PKG”)。PKG 利用一个主密文信息来产生私有密钥。由于一个实体的公共密钥可根据其身份推知,因此当 Alice 想要向 Bob 发送一条消息时,她就不必从数据库中取回 Bob 的公共密钥。而是 Alice 只需根据 Bob 的识别信息直接推知密钥。公共密钥数据库就成为多余,认证授权(“CAs”)也是不必要的了。无需将 Bob 的身份“绑定”到他的公共密钥上,因为他的身份即是他的公共密钥。

[0006] 基于身份的系统的概念并不新鲜。它在 A. Shamir 所著的“Identity-Based Cryptosystems and Signatures Schemes(基于身份的加密系统与签名方案)”中就已经被提出,该文发表于 ADVANCES IN CRYPTOGRAPHY-CRYPTO '84, Lecture Notes in Computer Science 196(1984), Springer, 47-53。然而,可实际应用的基于身份的加密方案至今仍未被找到。例如,基于身份的方案在下列文献中就已经被提出, C. Cocks 所著的“An Identity-Based Encryption Scheme Based on Quadratic Residues(以平方残差为基础的基于身份的加密方案)”,该文可在 <http://www.cesg.gov.uk/technology/id-pkc/media/ciren.pdf> 得到; D. Boneh, M. Franklin 所著的“Identity Based Encryption from the Weil Pairing(由 Weil 配对得到的基于身份的加密)”,该文发表于 ADVANCES IN CRYPTOGRAPHY-CRYPTO2001, Lecture Notes in Computer Science 2139(2001), Springer, 213-229; 以及 D. Boneh, M. Franklin 所著的“Identity Based Encryption from the Weil Pairing(extended version)(由 Weil 配对得到的基于身份的加密(扩展版本))”,该文可在 <http://www.cs.stanford.edu/~dabo/papers/ibe.pdf> 得到。Cocks 的方案是基于“平方残差问题”的,尽管加密和解密都相当地快(大约是 RSA 的速度),但是会有显著的消

息扩展（即密文比特长度是明文比特长度的许多倍）。Boneh-Franklin 方案将其安全性的基础建立在“双线性 Diffie-Hellman 问题”上，在使用超奇异椭圆曲线或阿贝尔簇曲线上的 Weil 或 Tate 配对时，该方案相当快速与高效。

[0007] 然而，已知的基于身份的加密方案都有一个显著的缺陷——它们都不是分级结构的。在非基于身份的公共密钥加密技术中，已经可以设置 CA 的分级结构，在该结构中，根 CA 可以为其他 CA 发放证书，而后者又可以为特定域内的用户发放证书。这样做是很值得的，因为它减轻了根 CA 的工作量。可供基于身份的加密技术使用的实用分级方案还未被开发出来。

[0008] 理想情况下，基于身份的分级加密方案将包括一个逻辑或实际 PKG 的分级结构。例如，一个根 PKG 可以向其他 PKG 发放私有密钥，而后者又可以向特定域内的用户发放私有密钥。同时，只要发送方获得了根 PKG 的公共参数，即使发送方根本不在系统中，也可以在不上线查找接收方的公共密钥或低级公共参数的情况下发送一条加密信息。基于身份的分级加密方案的另一个优点在于损坏控制。例如，一个域 PKG 的密文的泄漏并不会危及更高层次 PKG 的密文，也不会危及任何其他不是这个被损害的域 PKG 的直接下级的 PKG 的密文。而 Cocks 和 Boneh-Franklin 所提倡的方案并不具有这些特性。

[0009] 安全而实用的基于身份的分级加密方案还未被开发出来。一种具有部分共谋抵抗性的基于身份的分级密钥共享方案已经在下列文献中被提出：G. Hanaoka, T. Nishioaka, Y. Zheng, H. Imai 所著的“An Efficient Hierarchical Identity-Based Key-Sharing Method Resistant Against Collusion Attacks（一种能够抵抗共谋攻击的高效的基于身份的分级密钥共享方法）”，该文发表于 ADVANCES IN CRYPTOGRAPHY-ASIACRYPT 1999, Lecture Notes in Computer Science 1716(1999), Springer, 348-362；以及 G. Hanaoka, T. Nishioaka, Y. Zheng, H. Imai 所著的“A Hierarchical Non-Interactive Key-Sharing Scheme With Low Memory Size and High Resistance Against Collusion Attacks（一种具有低内存容量、高共谋攻击抵抗性的分级非互动密钥共享方案）”，该文将发表于 THE COMPUTER JOURNAL。另外，在 J. Horwitz, B. Lynn 所著的“Toward Hierarchical Identity-Based Encryption（走向基于身份的分级加密）”一文中还提供了对基于身份的分级加密的介绍，该文即将发表于 ADVANCES IN CRYPTOGRAPHY-EUROCRYPT 2002, Lecture Notes in Computer Science, Springer。Horwitz 和 Lynn 提出了一种两级的分级方案，该方案在第一级具有完全的共谋抵抗性，并在第二级具有部分的共谋抵抗性（即，用户可以共谋以获得他们的域 PKG 的密文，并据此假扮成域 PKG）。但是，Horwitz-Lynn 系统的复杂性会随着第二级上的共谋抵抗性提高，因此该方案不可能做到既实用又安全。

[0010] 因此需要一种安全实用的基于身份的分级加密方案。本发明的一个目标就是要提供一种安全而又实用的基于身份的分级加密方案。本发明的另一个目标是要提供一种安全而又实用的基于身份的分级签名方案。本发明的另一个目标是所述的加密与签名方案都是完全可调整的。本发明的另一个目标是所述的加密与签名方案在任意数量的级别上都具有完全的共谋抵抗性，并且它们具有随机预言模型中的选定密文安全性。

发明内容

[0011] 根据本发明，它提供了用来实现安全可靠且实用的基于身份的分级加密与签名方

案的方法。

[0012] 根据本发明的一方面内容,它提供了一种用来在一个系统中的发送方与接收方之间编码和解码数字消息的方法,所述的系统中包括多个私有密钥生成器 (“PKG”)。这些 PKG 中至少包括一个根 PKG 以及根 PKG 与接收方之间的分级结构中的 n 个低级 PKG, 其中 $n \geq 1$ 。一个根密钥生成密文被选取, 并且仅为根 PKG 所知。根据所述的根密钥生成密文产生一个根密钥生成参数。为 n 个低级 PKG 各选取一个低级密钥生成密文, 其中每个低级密钥生成密文仅对其相关的低级 PKG 已知。还要为 n 个低级 PKG 各产生一个低级密钥生成参数, 其中至少要用到对应于相关低级私有密钥生成器的低级密钥生成密文。至少利用根密钥生成参数和接收方身份信息来对所述的消息进行编码, 以形成一个密文。产生一个接收方私有密钥, 使得该接收方私有密钥至少与根密钥生成密文、根 PKG 和接收方之间的分级结构中的 n 个低级 PKG 相关的 n 个低级密钥生成密文中的一个或多个、以及接收方身份信息有关。至少利用接收方私有密钥来解密所述的密文以恢复所述的消息。

[0013] 根据本发明的另一方面内容,它提供了一种用来在一个系统中的发送方与接收方之间编码和解码数字消息的方法,所述的系统中包括多个私有密钥生成器 (“PKG”)。这些 PKG 中至少包括一个根 PKG、根 PKG 与发送方之间的分级结构中的 m 个低级 PKG, 其中 $m \geq 1$, 根 PKG 与接收方之间的分级结构中的 n 个低级 PKG, 其中 $n \geq 1$, 以及 PKG_1 , 它是发送方与接收方共同的前辈 PKG。在该分级结构中, m 个私有密钥生成器中的 1 个是发送方与接收方共同的前辈 PKG, 其中 $1 \geq 1$ 。

[0014] 根据本发明这一方面的内容,要为根 PKG 与发送方之间的分级结构中的 m 个低级 PKG 各选取一个低级密钥生成密文。生成一个发送方私有密钥, 使得该发送方私有密钥至少与根密钥生成密文、根 PKG 和发送方之间的分级结构中的 m 个低级 PKG 相关的 m 个低级密钥生成密文中的一个或多个、以及发送方身份信息有关。生成一个接收方私有密钥, 使得该接收方私有密钥至少与根密钥生成密文、根 PKG 和接收方之间的分级结构中的 n 个低级 PKG 相关的 n 个低级密钥生成密文中的一个或多个、以及接收方身份信息有关。至少利用接收方身份信息、发送方私有密钥、以及和位于或低于公共前辈 PKG_1 级别的 $(m-1+1)$ 个私有密钥生成器相关的低级密钥生成参数中的 0 个或多个, 来对所述的消息进行编码, 但是不能使用和高于公共前辈 PKG_1 的 $(1-1)$ 个 PKG 相关的低级密钥生成参数中的任何一个。至少利用发送方身份信息、接收方私有密钥、以及和位于或低于公共前辈 PKG_1 级别的 $(n-1+1)$ 个私有密钥生成器相关的低级密钥生成参数中的 0 个或多个, 来对所述的消息进行解码, 但是不能使用和高于公共前辈 PKG_1 的 $(1-1)$ 个 PKG 相关的低级密钥生成参数中的任何一个。

[0015] 根据本发明的另一方面内容,它提供了一种用来在一个系统中的发送方与接收方之间产生及验证一条消息的数字签名的方法,所述的系统中包括多个 PKG。这些 PKG 中至少包括一个根 PKG 以及根 PKG 与发送方之间的分级结构中的 n 个低级 PKG, 其中 $n \geq 1$ 。一个根密钥生成密文被选取, 并且仅为根 PKG 所知。根据所述的根密钥生成密文产生一个根密钥生成参数。为 n 个低级 PKG 各选取一个低级密钥生成密文, 其中每个低级密钥生成密文仅对其相关的低级 PKG 已知。还要为 n 个低级 PKG 各产生一个低级密钥生成参数, 其中至少要用到对应于相关低级私有密钥生成器的低级密钥生成密文。为发送方生成一个私有密钥, 使得该私有密钥至少与根密钥生成密文以及发送方身份信息有关。至少利用发送方私

有密钥来签署所述的消息以生成一个数字签名。至少利用根密钥生成参数以及发送方身份信息来验证所述的数字消息。

附图说明

[0016] 以下对本发明优选实施例的说明参照了附图,其中:

[0017] 图 1 示出了一张流程图,该图根据本发明的当前优选实施例展示了一种编码和解码数字消息的方法;

[0018] 图 2 示出了一张流程图,该图根据本发明的另一个当前优选实施例展示了一种在发送方 y 和接收方 z 之间编码及解码数字消息的方法;

[0019] 图 3 示出了一张框图,该图展示了一种典型的分级结构,在这种结构中可以实现图 2 所示的方法;

[0020] 图 4 示出了一张流程图,该图根据本发明的另一个当前优选实施例展示了一种编码和解码一条数字消息 M 的方法,所述的数字消息在发送方 y 和接收方 z 之间传递;

[0021] 图 5 示出了一张流程图,该图根据本发明的另一个当前优选实施例展示了一种编码和解码一条数字消息 M 的方法,所述的数字消息在发送方 y 和接收方 z 之间传递;

[0022] 图 6 示出了一张流程图,该图根据本发明的另一个当前优选实施例展示了一种编码和解码一条数字消息 M 的方法,所述的数字消息在发送方 y 和接收方 z 之间传递;

[0023] 图 7 示出了一张流程图,该图根据本发明的另一个当前优选实施例展示了一种生成及验证一个数字签名的方法;

[0024] 图 8 示出了一张流程图,该图根据本发明的另一个当前优选实施例展示了一种生成及验证一条数字消息 M 的数字签名 sig 的方法,所述的数字消息在发送方 y 和接收方 z 之间传递;以及

[0025] 图 9 示出了一张流程图,该图根据本发明的另一个当前优选实施例展示了一种生成及验证一条数字消息 M 的数字签名 sig 的方法,所述的数字消息在发送方 y 和接收方 z 之间传递。

具体实施方式

[0026] 本发明的当前优选方法提供了安全可靠而实用的基于身份的分级加密 (“HIDE”) 与签名 (“HIDS”) 方案。所述的分级方案是完全可调整的,在任意数量的级别上都具有完全的共谋抵抗性,并且具有随机预言模型中的选定密文安全性。这些目标部分是通过在各个低级 PKG 中引入附加随机信息而实现的。这些方案在直观上令人惊讶的一个方面在于,即使低级 PKG 产生了附加随机信息,也不会迫使在分级结构的根级别以下添加公共参数。另外,低级 PKG 所产生的随机信息不会对不在低级 PKG 以下的用户向低级 PKG 以下的用户发送加密信息的能力造成负面影响。

[0027] 本发明的每一个 HIDE 和 HIDS 方案都需要 PKG 的分级结构,该结构中至少包括一个根 PKG 和多个低级 PKG。分级结构和低级 PKG 可以是逻辑的,也可以是实际的。例如,一个单个实体就可以产生根密钥生成密文和低级密钥生成密文,低级用户的加密或签名密钥都是由后者生成的。在这种情况下,低级 PKG 都不是独立的实体,而只是以逻辑分级结构组织的进程或信息,并被用来为分级结构中的后代 PKG 及用户生成密钥。或者,每个低级 PKG

也可以是独立的实体。另一种备选方案涉及实际与逻辑低级 PKG 的混合形式。为了本文公开说明的目的,短语“低级 PKG”一般被用来指代这些备选方案中的任意一种。

[0028] 在本文所公开的基于身份的分级密码系统环境下,基于身份的公共密钥可以是基于时间周期的。例如,一个特定接收方的身份会随各个随后的时间周期而变化。或者,接收方也可以将时间周期安排为它自己在分级结构中的后代或下级,并且发送方会在编码消息时使用正确时间周期上的身份。不论采用何种方式,每个密钥都只在相关的时间周期内才能有效的用来签署要送给 Bob 的消息。

[0029] 本发明的 HIDE 方案通常包括 5 个随机化的算法:根设置、低级设置、抽取、加密以及解密。这些算法中的三个依赖于分级结构中相关实体的身份。每名用户最好都在分级结构中拥有一个位置,该分级结构可由它的 ID 元组 $(ID_1, \dots, ID_t)$ 定义。所述用户在分级结构中的前辈是根 PKG 以及 ID 元组为 $\{(ID_1, \dots, ID_i) : 1 \leq i \leq (t-1)\}$ 的用户或 PKG。为了计算的目的, ID 元组最好用二进制字符串表示。

[0030] 在根设置算法中,根 PKG 使用一个保密参数 k 来产生公共系统参数 $params$ 以及一个根密钥生成密文。所述的系统参数包括对消息空间 M 和密文空间 X 的描述。所述的系统参数是公开使用的,但只有根 PKG 知道根密钥生成密文。

[0031] 在低级设置算法中,为了抽取的目的,每个低级 PKG 最好产生它自己的低级密钥生成密文。或者,低级 PKG 也可以为每次抽取产生一次性密文。

[0032] 在抽取算法中,一个 PKG (根 PKG 或低级 PKG) 为它的任意一个后代产生一个私有密钥。该私有密钥是利用系统参数、生出方 PKG 的私有密钥以及任何其他的优选密文信息产生的。

[0033] 在加密算法中,发送方从根 PKG 接收系统参数,最好是通过本系统以外的某些安全途径接收。发送方不必接收任何低级密钥生成参数。所述的发送方利用 $params$ 和期望接收方的 ID 元组来编码一条消息 $M \in M$, 以产生一个加密文本 $C \in X$ 。相反地,在解码算法中,接收方利用 $params$ 和接收方的私有密钥 d 来解码加密文本 C , 以恢复消息 M 。加密和解密最好都满足标准的一致性约束:

[0034] $\forall M \in M : \text{Decryption}(params, d, C) = M$

[0035] 其中 $C = \text{Encryption}(params, \text{ID 元组}, M)$ 。

[0036] 象 HIDE 方案一样,本发明的 HIDS 方案一般也包括 5 个随机化的算法:根设置、低级设置、抽取、签署以及验证。在根设置中,系统参数会被补充以包括进对签名空间 Σ 的说明。低级设置和抽取最好与上述 HIDE 中的算法一样。

[0037] 在签署算法中,数字消息的发送方利用 $params$ 和发送方的私有密钥 d 签署消息 $M \in M$, 以生成一个签名 $S \in \Sigma$ 。在验证算法中,被签署消息的接收方利用 $params$ 以及发送方的 ID 元组来验证签名 S 。验证算法最好输出“有效”或“无效”。签署与验证最好也满足一致性约束:

[0038] $\forall M \in M : \text{Verification}(params, \text{ID 元组}, S) = \text{“有效”}$

[0039] 其中 $S = \text{Signing}(params, d, M)$ 。

[0040] HIDE 与 HIDS 方案的安全性

[0041] 下面将分别针对 HIDE 和 HIDS 来说明实现了本发明的方案的安全性。在基于身份的非分级加密技术环境下人们已经注意到,必须为基于身份的系统加强选定密文安全性的

标准定义。这是因为,为了进行安全性分析,应该假定一个敌对方能够获得与其选择的任意身份相关的私有密钥(除了受到攻击的特定身份以外)。这一点同样适用于基于身份的分级加密技术。因此,为了确保本发明的 HIDE 方案是具有选定密文安全的,就可以让一个模拟攻击者进行密钥抽取查询。同时,还要允许该模拟敌对方选择其所希望挑战的身份。

[0042] 还应该注意的是,一个敌对方可以自适应性或非自适应地选取其对象的身份。自适应地选取其对象的敌对方将首先进行乱序查询和抽取查询,并接着根据这些查询的结果选取它的目标。这样的敌对方在开始攻击时不会有特定的目标。而是,只要它能破解某人,该敌对方就是成功的。另一方面,一个非自适应的敌对方不会根据乱序查询和抽取查询的结果来选取它的目标。例如,这样的敌对方会以一个私敌为目标。该敌对方仍会进行乱序查询和抽取查询,但是它的目标选择是严格根据目标身份的,而非根据查询结果。显而易见,针对自适应目标选取的敌对方的安全性是更强的,因此也是更加优越可取的安全性概念。但是,对本发明中 HIDE 方案的安全性分析提到了两种类型的安全性。

[0043] 如果在以下的竞赛中不存在对挑战者具有不可忽略优势的受多项式限制的敌对方 A,那么就称 HIDE 方案对自适应选取密文以及自适应选取目标的攻击具有语义上的安全性。

[0044] 设置:挑战者取得一个保密参数 k 并运行根设置算法。它将得到的系统参数 params 提供给敌对方。它将根密钥生成密文留给自己。

[0045] 阶段 1:敌对方提出查询 $q_1, \dots, q_m$, 其中 q_i 是下列查询中的一种:

[0046] 1. 公共密钥查询 (ID 元组 i):挑战者对 ID 元组 i 运行一个乱序算法,以获得对应于 ID 元组 i 的公共密钥 $H(\text{ID 元组 } i)$ 。

[0047] 2. 抽取查询 (ID 元组 i):挑战者运行抽取算法以生成对应于 ID 元组 i 的私有密钥 d_i , 并将 d_i 发送给敌对方。

[0048] 3. 解密查询 (ID 元组 i, C_i):挑战者运行抽取算法以生成对应于 ID 元组 i 的私有密钥 d_i , 利用 d_i 运行解密算法以解密 C_i , 并将结果得到的明文发送给敌对方。

[0049] 这些查询可以自适应地提出。另外,被查询的 ID 元组 i 可以对应于分级结构任何一个级别上的位置。

[0050] 挑战:一旦敌对方判定阶段 1 已经结束,它就会输出两个长度相等的明文 $M_0, M_1 \in M$, 以及一个它希望挑战的 ID 元组。唯一的限制在于,该 ID 元组以及它的前辈都不能出现在阶段 1 中的任何私有密钥抽取查询中。挑战者随意选取一个随机比特 $b \in \{0, 1\}$, 并设 $C = \text{Encryption}(\text{params}, \text{ID 元组}, M_b)$ 。它将 C 作为一次挑战发送给敌对方。

[0051] 阶段 2:敌对方提出更多的查询 $q_{m+1}, \dots, q_n$, 其中 q_i 是下列查询中的一种:

[0052] 1. 公共密钥查询 (ID 元组 i):挑战者像在阶段 1 中那样进行回应。

[0053] 2. 抽取查询 (ID 元组 i):挑战者像在阶段 1 中那样进行回应。

[0054] 3. 解密查询 ($C, \text{ID 元组 } i$):挑战者像在阶段 1 中那样进行回应。

[0055] 阶段 2 中的查询受到如下限制,即挑战者不能对与挑战密文 C 相关的 ID 元组进行抽取查询,或是利用那个 ID 元组以及密文 C 进行解密查询。这一限制同样适用于该 ID 元组的所有前辈。

[0056] 猜测:敌对方输出一个猜测值 $b' \in \{0, 1\}$ 。如果 $b = b'$, 则敌对方赢得比赛。敌对方在攻击本方案中所拥有的优势被定义为 $|\Pr[b = b'] - 1/2|$ 。

[0057] 在以下所述的竞赛中,如果不存在拥有不可忽略优势的多项式时间敌对方,那么 HIDE 方案就被称为单向加密方案。在该竞赛中,敌对方 A 被给予一个随机公共密钥 K_{pub} 和一个加密文本 C,并输出一个对明文的猜测值,所述的加密文本 C 是利用 K_{pub} 对随机消息 M 进行加密得到的。如果 ϵ 是 A 输出 M 的概率,那么就称所述的敌对方对本方案具有优势 ϵ 。所述的竞赛如下进行:

[0058] 设置:挑战者取得一个保密参数 k 并运行根设置算法。它将得到的系统参数 params 提供给敌对方。它将根密钥生成密文留给自己。

[0059] 阶段 1:敌对方就如在上述选定密文安全性分析的阶段 1 中那样进行公共密钥和/或抽取查询。

[0060] 挑战:一旦敌对方判定阶段 1 已经结束,它就会输出一个它希望挑战的新 ID 元组 ID。挑战者随意选取一个随机的 $M \in M$,并设 $C = \text{Encryption}(\text{params}, \text{ID 元组}, M)$ 。它将 C 作为一次挑战发送给敌对方。

[0061] 阶段 2:敌对方对除了 ID 及其前辈以外的其他身份提出更多的公共密钥查询以及更多的抽取查询,挑战者则会如阶段 1 中那样作出回应。

[0062] 猜测:敌对方输出一个猜测值 $M' \in M$ 。如果 $M = M'$,则敌对方赢得比赛。敌对方在攻击本方案中所拥有的优势被定义为 $\Pr[M = M']$ 。

[0063] 本发明的方案对于上述挑战是安全可靠的。另外,本发明的 HIDS 方案针对现有的针对自适应选取消息的伪造也是安全可靠的。即使在(自适应地)获取了目标在敌对方所选取的消息上的签名之后,敌对方也不能伪造出它的目标在其以前并未签署过的其他消息上的签名。一个 HIDS 敌对方还将拥有对除了它的目标及其前辈以外的其他实体进行公共密钥查询和私有密钥抽取查询的能力,以及选取其目标的能力。对于 HIDE 而言,敌对方的目标选择可以是自适应的也可以是非自适应的。

[0064] 配对

[0065] 本发明的当前优选 HIDE 和 HIDS 方案都是基于配对的,例如与椭圆曲线或阿贝尔簇曲线相关的 Weil 或 Tate 配对。所述的方法也可以是基于双线性 Diffie-Hellman 问题的。它们使用两个循环群 Γ_1 和 Γ_2 ,这两个循环群最好具有同样大小的素数阶 q。第一群 Γ_1 最好是椭圆曲线或阿贝尔簇曲线上的一群点,并且 Γ_1 上的群规则可以被写成加性的。第二群 Γ_2 最好是一个有限域的乘性子群,并且 Γ_2 上的群规则可以被写成乘性的。但是,也可以使用其他类型的群作为符合本发明的 Γ_1 和 Γ_2 。

[0066] 所述的方法还利用了第一群 Γ_1 的生成器 P_0 。另外,还提供了一个配对或函数 $\hat{e}: \Gamma_1 \times \Gamma_1 \rightarrow \Gamma_2$,用来将第一群 Γ_1 的两个元素映射成第二群 Γ_2 的一个元素。函数 $\hat{e}$ 最好满足三个条件。首先,函数 $\hat{e}$ 最好是双线性的,如果 Q 和 R 都在 Γ_1 中,且 a 和 b 都是整数,那么 $\hat{e}(aQ, bR) = \hat{e}(Q, R)^{ab}$ 。第二,函数 $\hat{e}$ 最好是非退化的,从而使得该映射不会将 $\Gamma_1 \times \Gamma_1$ 中的所有配对转变为 Γ_2 中的身份。第三,函数 $\hat{e}$ 最好是可以高效计算的。满足这三个条件的函数 $\hat{e}$ 被认为是可行的。

[0067] 函数 $\hat{e}$ 最好还是对称的,从而对所有的 $Q, R \in \Gamma_1$ 都有 $\hat{e}(Q, R) = \hat{e}(R, Q)$ 。然而,对称性直接来自于双线性以及 Γ_1 是循环群这样一个事实。可根据现有技术中已知的方法来修改与超奇异椭圆曲线以及阿贝尔簇曲线相关的 Weil 和 Tate 配对,以创建这样的双线性映射。但是,即使将第一循环群 Γ_1 的元素称为“点”会暗示函数 $\hat{e}$ 是一种经过修改的

Weil 或 Tate 配对,但应该注意的是,任何可行的配对 $\hat{e}$ 都能够发挥作用。

[0068] 本发明中 HIDE 和 HIDS 方案的安全性主要是基于双线性 Diffie-Hellman 问题的。双线性 Diffie-Hellman 问题是在给定一个随机选取的 $P \in \Gamma_1$ 以及 aP 、 bP 和 cP (对于未知的随机选取的 $a, b, c \in \mathbb{Z}/q\mathbb{Z}$) 的情况下,求出 $\hat{e}(P, P)^{abc}$ 的问题。在 Γ_1 中解决 Diffie-Hellman 问题就解决了双线性 Diffie-Hellman 问题,这是因为 $\hat{e}(P, P)^{abc} = \hat{e}(abP, cP)$ 。类似地,在 Γ_2 中解决 Diffie-Hellman 问题也就解决了双线性 Diffie-Hellman 问题,因为如果 $g = \hat{e}(P, P)$, 那么 $g^{abc} = (g^{ab})^c$, 其中 $g^{ab} = \hat{e}(aP, bP)$ 且 $g^c = \hat{e}(P, cP)$ 。为了使双线性 Diffie-Hellman 问题变得困难,就应该对 Γ_1 和 Γ_2 进行选取,使得在 Γ_1 或 Γ_2 中不存在能够有效地解决 Diffie-Hellman 问题的已知算法。

[0069] 如果一个随机化算法 $I\Gamma$ 采用了一个保密参数 $k > 0$ 、在 k 的多项式时间内运行、并输出两个群 Γ_1 和 Γ_2 的描述以及一个可行配对 $\hat{e} : \Gamma_1 \times \Gamma_1 \rightarrow \Gamma_2$ 的描述,其中所述的两个群最好具有相同的素数阶 q , 那么该算法 $I\Gamma$ 就是一个双线性 Diffie-Hellman 生成器。如果 $I\Gamma$ 是一个双线性 Diffie-Hellman 参数生成器, 那么一个算法 B 在解决双线性 Diffie-Hellman 问题中所拥有的优势 $\text{Adv}_{I\Gamma}(B)$ 就被定义为, 当送入算法的输入项为 Γ_1 、 Γ_2 、 $\hat{e}$ 、 P 、 aP 、 bP 和 cP 时, 算法 B 输出 $\hat{e}(P, P)^{abc}$ 的概率, 其中 $(\Gamma_1, \Gamma_2, \hat{e})$ 是 $I\Gamma$ 针对足够大的保密系数 k 的输出, P 是 Γ_1 的随机生成器, a 、 b 和 c 则是 $\mathbb{Z}/q\mathbb{Z}$ 的随机元素。双线性 Diffie-Hellman 问题下的假设是, $\text{Adv}_{I\Gamma}(B)$ 对于所有有效算法 B 都是可忽略的。

[0070] HIDE 方案

[0071] 现在参见附图, 图 1 所示的流程图根据本发明的一种当前优选实施例展示了一种编码和解码数字消息的方法。该方法在包括多个 PKG 的 HIDE 系统中执行。所述的 PKG 中至少包括一个根 PKG 以及根 PKG 和接收方之间的分级结构中的 n 个低级 PKG, 其中 $n \geq 1$ 。

[0072] 在模块 102 中, 根 PKG 选取一个只有根 PKG 知道的根密钥生成密文。该根密钥生成密文可被用来为分级结构中根 PKG 以下的 PKG 和 / 或用户生成私有密钥。然后, 在模块 104 中, 根 PKG 根据根密钥生成密文产生一个根密钥生成参数。该根密钥生成参数被用来掩饰根密钥生成密文。该根密钥生成参数可被透露给低级 PKG, 而又不会危及根密钥生成密文。在模块 106 中, 低级 PKG 选取低级密钥生成密文。与一个给定的低级 PKG 相关的低级密钥生成密文可被用来为分级结构中在该相关低级 PKG 之下的 PKG 和 / 或用户生成私有密钥。与根密钥生成密文类似, 每个低级密钥生成密文仅对其相关的低级 PKG 已知。

[0073] 在模块 108 中, 为 n 个低级 PKG 各自产生低级密钥生成参数。每个低级密钥生成参数的产生至少要利用其相关低级 PKG 的低级密钥生成密文。与根密钥生成参数类似, 每个低级密钥生成参数掩饰了与其相关的低级密钥生成密文。

[0074] 在模块 110 中, 发送方至少利用根密钥生成参数以及与接收方相关的身份信息来编码消息以形成一个密文。例如, 可以只利用根密钥生成参数以及接收方的身份来编码所述的消息。或者, 也可以利用低级密钥生成参数中的一个, 就像在下文中将根据双 HIDE 方案进行更详细说明的一样。在模块 112 中, 一个低级 PKG 为接收方生成一个私有密钥, 使得该私有密钥至少与根密钥生成密文、与分级结构中根 PKG 和接收方之间的 n 个低级 PKG 相关的 n 个低级密钥生成密文中的一个或多个、以及接收方的身份信息相关。例如, 除了根密钥生成密文和接收方的身份信息之外, 接收方的私有密钥最好还至少与向接收方发放私有密钥的 PKG 的低级密钥生成密文相关。或者, 接收方的私有密钥也可以与所有 n 个前辈 PKG

的低级密钥生成密文以及根密钥生成密文相关。在模块 114 中,接收方至少利用其私有密钥来解码密文并恢复消息。除了利用其私有密钥来解码以外,接收方最好还利用与分级结构中根 PKG 和接收方之间的 n 个低级 PKG 相关的 n 个低级密钥生成参数。

[0075] 每个低级 PKG 都有一个密钥生成密文,就像根 PKG 一样。如上所述,低级 PKG 最好利用这个密文来为它的各个后代生成私有密钥,就像根 PKG 那样做。这样,后代的私有密钥就与低级 PKG 的密钥生成密文相关。即使低级 PKG 为了限制密钥托管 (escrow) 的目的而使用其密钥生成密文的修改版本来隐藏那个密文,这一点也是成立的,正如下文中将要更完整地说明的那样。同时,低级 PKG 不必总是使用相同的密文来进行每次私有密钥提取,而是可以为 PKG 的每个后代随机产生一个新的密钥生成密文,从而为每个后代得到不同的密钥生成参数。

[0076] 由于一个低级 PKG 能够为接收方生成一个私有密钥 (模块 112),因此根 PKG 不必自己生成所有的私有密钥。另外,由于低级 PKG 使用它们自己的密钥生成密文来为它们的后代产生私有密钥,因此暴露一个低级密钥生成密文只会对分级结构造成有限的安全性损害。与其暴露分级结构中所有的私有密钥,不如让一个低级 PKG 的一次违规行为只暴露该 PKG 的私有密钥以及利用那个 PKG 的密钥生成密文产生的那些私有密钥 (即,作为被暴露 PKG 在分级结构中的直系后代的那些用户的私有密钥)。

[0077] 本实施例的另一个优点在于,发送方不必处在分级结构中即可向接收方发送一个编码消息。该发送方只需知道与接收方相关的身份信息以及由根 PKG 生成的系统参数。但是,当发送方处于分级结构中时,本发明的 HIDE 方案还有某些额外的优点会体现出来。例如,当发送方与接收方都处在分级结构中时,就可以通过利用双方的身份来改善消息加密的效率。这类 HIDE 方案可以被称为双 HIDE,因为发送方与接收方的身份都被用作加密及解密算法的输入。现在将参照图 2 和图 3 来说明使用了双 HIDE 方案的编码与解码方法。

[0078] 双 HIDE

[0079] 图 2 所示的流程图根据本发明的另一个当前优选实施例展示了一种在发送方 y 和接收方 z 之间编码及解码数字消息的方法。图 3 所示的框图展示了一种典型的分级结构,在这种结构中可以实现这种方法。与先前的实施例相似,该方法可在一个 HIDE 系统中实现,所述的 HIDE 系统中至少包括一个根 PKG 302,以及根 PKG 302 与接收方 z 308 之间的分级结构中的 n 个低级 PKG 304a, b, d, 其中 $n \geq 1$ 。该实施例中的发送方 y 306 必须也在分级结构中,并且该分级结构中还包括根 PKG 302 与发送方 y 306 之间的 m 个低级 PKG 304a, b, c, 其中 $m \geq 1$ 。在根 PKG 302 与发送方 y 306 之间的 m 个 PKG 304a, b, c, 以及根 PKG 302 与接收方 z 308 之间的 n 个 PKG 304a, b, d 中,有 1 个 PKG 304a, b 是发送方 y 306 与接收方 z 308 的公共前辈,其中 $1 \leq l \leq m, n$ 。例如,在图 3 中示出了这 1 个公共前辈 PKG 中的两个 ($\text{PKG}_{y1}/\text{PKG}_{z1}$ 304a 和 $\text{PKG}_{y1}/\text{PKG}_{z1}$ 304b)。

[0080] 该实施例的方法在模块 202 中开始,在该模块中,根 PKG 302 选取一个只有根 PKG 302 知道的根密钥生成密文。然后,在模块 204 中,根 PKG 302 根据根密钥生成密文产生一个根密钥生成参数。在模块 206 中,低级 PKG 304a-d 选取低级密钥生成密文。与根密钥生成密文类似,每个低级密钥生成密文只对与其相关的低级 PKG 304a-d 已知。在模块 208 中,为 n 个低级 PKG 304a-d 各自产生低级密钥生成参数。每个低级密钥生成参数的产生至少要用到对应于其相关低级 PKG 304a-d 的低级密钥生成密文。

[0081] 在模块 210 中,发送方的上代 PKG_{ym} 304c 为发送方 y 306 生成一个私有密钥,使得该私有密钥至少与根密钥生成密文、与根 PKG 302 和发送方 y 306 之间的 m 个低级 PKG 304a, b, c 相关的 m 个低级密钥生成密文中的一个或多个、以及发送方的身份信息相关。例如,除了根密钥生成密文和发送方的身份信息以外,发送方的私有密钥最好至少还与发送方的上代 PKG_{ym} 304c 的低级密钥生成密文相关。或者,发送方的私有密钥也可以与它所有 m 个直系前辈 PKG 的低级密钥生成密文以及根密钥生成密文相关。在模块 212 中,接收方的上代 PKG_{zn} 304d 为接收方 z 生成一个私有密钥,生成的方式与发送方的上代 PKG_{ym} 304c 用来生成发送方私有密钥的方式类似。

[0082] 在模块 214 中,发送方 y 编码消息以形成一个密文,该过程至少要用到发送方的私有密钥以及与根 PKG 302 和发送方 y 306 之间的 $(m-1+1)$ 个 PKG (即, PKG_{y1} 304b 和 PKG_{ym} 304c) 相关的低级密钥生成参数中的一个或多个,且所述的 PKG 位于发送方 y 306 和接收方 z 308 的最低级公共前辈 $\text{PKG}(\text{PKG}_{y1}/\text{PKG}_{z1}$ 304b) 的级别或低于该级别。在对消息进行编码时,发送方 y 306 最好不要用到与高于最低级公共前辈 $\text{PKG}(\text{PKG}_{y1}/\text{PKG}_{z1}$ 304b) 的 $(1-1)$ 个 PKG (即 PKG_{y1} 304a) 有关的任何低级密钥生成参数。

[0083] 然后,在模块 216 中,接收方 z 308 解码密文以恢复出所述的消息,该过程至少要用到接收方的私有密钥以及与根 PKG 302 和接收方 z 308 之间的 $(n-1+1)$ 个 PKG (即, PKG_{z1} 304b 和 PKG_{zn} 304c) 相关的低级密钥生成参数中的一个或多个,且所述的 PKG 位于发送方 y 306 和接收方 z 308 的最低级公共前辈 $\text{PKG}(\text{PKG}_{y1}/\text{PKG}_{z1}$ 304b) 的级别或低于该级别。在对消息进行解码时,接收方 z 308 最好不要用到与高于最低级公共前辈 $\text{PKG}(\text{PKG}_{y1}/\text{PKG}_{z1}$ 304b) 的 $(1-1)$ 个 PKG (即 PKG_{z1} 304a) 有关的任何低级密钥生成参数。

[0084] 本发明的这种双 HIDE 实施例提供了对消息进行编码和解码的更为高效的方案,因为只需使用较少的密钥生成参数。例如,普通 HIDE 方案中的解码大约需要所有的 n 个密钥生成参数,但是在双 HIDE 方案中的解码只需要 $(n-1+1)$ 个密钥生成参数。双 HIDE 方案要求发送方 y 306 在向接收方 z 308 发送一条编码消息之前先获得它的私有密钥,这与只需获得根 PKG 的公共系统参数相反。双 HIDE 方案还使得发送方 y 306 和接收方 z 308 能够限制密钥托管的范围,就如下文中将要更完整说明的那样。这个共享的密文是除了它们最低级公共前辈 $\text{PKG}_{y1}/\text{PKG}_{z1}$ 304b 以外的第三方所不知道的。

[0085] 基本 HIDE

[0086] 图 4 所示的流程图根据本发明的另一个当前优选实施例展示了一种编码和解码一条数字消息 M 的方法,所述的数字消息在发送方 y 和接收方 z 之间传递。如图 3 中所示,接收方 z 308 在分级结构中比根 PKG 低 $n+1$ 个级别,并且与 ID 元组 $(\text{ID}_{z1}, \dots, \text{ID}_{z(n+1)})$ 相关联。接收方的 ID 元组中包括与接收方有关的身份信息 $\text{ID}_{z(n+1)}$,以及与其在分级结构中的 n 个前辈低级 PKG 相关的身份信息 ID_{z1} 。该方法在模块 402 中开始,在该模块中生成元素的第一与第二循环群 Γ_1 和 Γ_2 。在模块 404 中,选取一个函数 $\hat{e}$,使得该函数 $\hat{e}$ 能够由第一循环群 Γ_1 的两个元素生成第二循环群 Γ_2 的一个元素。函数 $\hat{e}$ 最好是一种可行的配对,如上所述。在模块 406 中选取第一循环群 Γ_1 的一个根生成器 P_0 。在模块 408 中,选取一个随机根密钥生成密文 s_0 ,该密文与根 PKG 302 相关且只有根 PKG 302 知道。 s_0 最好是循环群 Z/qZ 的一个元素。在模块 410 中产生一个根密钥生成参数 $Q_0 = s_0 P_0$ 。 Q_0 最好是第一循环群 Γ_1 的一个元素。在模块 412 中,选取一个第一函数 H_1 ,使得 H_1 能够由第一串二进制数生成

第一循环群 Γ_1 的一个元素。在模块 414 中选取一个第二函数 H_2 , 使得 H_2 能够由第二循环群 Γ_2 的一个元素生成第二串二进制数。模块 402 至 414 的功能都是上述 HIDE 根设置算法的组成部分, 并且最好都在大致相同的时刻完成。作为示例, 比如那些在 Boneh-Franklin 中公开的函数就可以被用作 H_1 和 H_2 。

[0087] 接下来的一系列模块 (模块 416 至 424) 示出了作为低级设置算法的组成部分而执行的功能。在模块 416 中, 为接收方的 n 个前辈低级 PKG 各生成一个公共元素 P_{zi} 。每个公共元素 $P_{zi} = H_1(ID_1, \dots, ID_{zi})$ 最好都是第一循环群 Γ_1 的一个元素, 其中 $1 \leq i \leq n$ 。尽管是以单个模块表示的, 但是所有公共元素 P_{zi} 的产生需要进行一段时间, 而非一次全部完成。

[0088] 为接收方的 n 个前辈低级 PKG 304a, b, d 各选取一个低级密钥生成密文 s_{zi} (模块 418)。该低级密钥生成密文 s_{zi} 最好是循环群 Z/qZ 的元素, 其中 $1 \leq i \leq n$, 并且每个低级密钥生成密文 s_{zi} 最好只有它相关的低级 PKG 知道。同样, 尽管是以单个模块表示, 但是所有低级密钥生成密文 s_{zi} 的选取需要进行一段时间, 而非一次全部完成。

[0089] 为发送方的 n 个前辈低级 PKG 各生成一个低级机密元素 S_{zi} (模块 420)。每个低级机密元素 $S_{zi} = S_{z(i-1)} + s_{z(i-1)}P_{zi}$ 最好是第一循环群 Γ_1 的一个元素, 其中 $1 \leq i \leq n$ 。尽管与公共元素 P_{zi} 以及密文 s_{zi} 一样都是以单个模块表示的, 但是所有机密元素 S_{zi} 的产生也需要进行一段时间, 而非一次全部完成。为了这些重复的密钥生成过程的缘故, 可将 S_0 定为 Γ_1 的身份元素。

[0090] 还要为接收方的 n 个前辈低级 PKG 各产生一个低级密钥生成参数 Q_{zi} (模块 422)。每个密钥生成参数 $Q_{zi} = s_{zi}P_0$ 最好都是第一循环群 Γ_1 的元素, 其中 $1 \leq i \leq n$ 。同样, 尽管以单个模块表示, 但是所有密钥生成参数 Q_{zi} 的产生也需要进行一段时间, 而非一次全部完成。

[0091] 随后两个模块 (模块 424 和 426) 的功能是作为上述抽取算法的一部分而执行的。在模块 424 中生成与接收方 z 相关的接收方公共元素 $P_{z(n+1)}$ 。该接收方公共元素, $P_{z(n+1)} = H_1(ID_{z1}, \dots, ID_{z(n+1)})$, 最好是第一循环群 Γ_1 的一个元素。然后在模块 426 中生成与接收方 z

有关的接收方机密元素 $S_{z(n+1)}$ 。该接收方机密元素,
$$S_{z(n+1)} = S_{zn} + s_{zn}P_{z(n+1)} = \sum_{i=1}^{n+1} s_{z(i-1)}P_{zi},$$

最好也是第一循环群 Γ_1 的一个元素。

[0092] 为方便起见, 第一函数 H_1 可被选为一种迭代函数, 从而可以按照例如 $H_1(P_{z(i-1)}, ID_{zi})$ 而非 $H_1(ID_1, \dots, ID_{zi})$ 来计算公共点 P_i 。

[0093] 图 4 中所示的最后两个模块 (模块 428 和 430) 代表了上述的加密与解密算法。在模块 428 中, 消息 M 被编码以生成一个加密文本 C 。该编码过程最好至少用到根密钥生成参数 Q_0 以及 ID 元组 $(ID_{z1}, \dots, ID_{z(n+1)})$ 。然后在模块 430 中解码加密文本 C 以恢复消息 M 。该解码过程最好至少用到低级密钥生成参数 Q_{zi} 以及接收方机密元素 $S_{z(n+1)}$, 其中 $1 \leq i \leq n$ 。

[0094] 图 4 中所示的模块不必完全顺次出现。例如, 一个知道接收方身份的发送方可以在接收方获得私有密钥之前对通信进行加密。

[0095] 现在将参照图 5 和图 6, 来详细说明在对消息 M 及密文 C 进行编码和解码中所提到的参数及元素的具体运用。图 5 所示的流程图根据本发明的另一个当前优选实施例展示了一种编码和解码一条数字消息 M 的方法, 所述的数字消息在发送方 y 和接收方 z 之间传递。

在这个被称为基本 HIDE 的方案中,根设置、低级设置以及抽取算法都与图 4 中模块 402 至 426 所示的实施例相同。图 5 所示的流程图展示了加密与解密算法,它在模块 528a 中从选取一个随机加密参数 r 开始。 r 最好是循环群 Z/qZ 中的一个整数。然后在模块 528b 中利用公式 $C = [U_0, U_2, \dots, U_{n+1}, V]$ 生成加密文本 C 。该加密文本 C 中包括元素 $U_i = rP_{zi}$, 其中 $i = 0$ 和 $2 \leq i \leq n+1$, 这些元素与接收方在分级结构中的位置有关。加密文本 C 的其他组成部分是加密形式的实际消息, $V = M \oplus H_2(g^r)$, 其中 $g = \hat{e}(Q_0, P_{z1})$ 。元素 g 最好是第二循环群 Γ_2 的成员。在消息被编码之后,可以根据基本 HIDE 解密算法对其进行解密,在该解密

算法中利用公式 $M = V \oplus H_2\left(\frac{\hat{e}(U_0, S_{z(n+1)})}{\prod_{i=2}^{n+1} \hat{e}(Q_{i-1}, U_i)}\right)$ 从加密文本 C 中恢复出消息 M (模块 530)。

[0096] 全 HIDE

[0097] 利用已知的方法来使得单向加密方案具有针对选定加密文本攻击的安全性,可以将基本 HIDE 方案转换为全 HIDE 方案,后者在随机预言模型中具有选定加密文本安全性。现在将参照图 6 来说明一种具有选定密文安全性的全 HIDE 方案。

[0098] 图 6 所示的流程图根据本发明的另一个当前优选实施例展示了一种编码和解码一条数字消息 M 的方法,所述的数字消息在发送方 y 和接收方 z 之间传递。在本发明的该实施例中,根设置、低级设置以及抽取算法都与参照图 4 所述的实施例相同,只是该实施例的根设置算法需要两个额外的函数。因此,图 6 所示的流程图从额外函数 (模块 615a 和 615b) 的选择开始,并继续进行加密与解密算法 (模块 628a 至 630d)。

[0099] 通过选择一个第三函数 H_3 (模块 615a) 和一个第四函数 H_4 (模块 615b) 来完成根设置算法。第三函数 H_3 最好能够由两串二进制数产生循环群 Z/qZ 的一个整数。第四函数 H_4 最好能够由另一个二进制串产生一个二进制串。

[0100] 加密算法从模块 628a 开始,该模块示出了随机二进制串 σ 的选取。然后,该随机二进制串 σ 被用来生成一个随机整数 $r = H_3(\sigma, M, W)$, 如模块 628b 所示。其中 W 是实际消息 M 的对称加密。该加密最好是利用对称加密算法 E 生成的,并用 $H_4(\sigma)$ 作为加密密钥。相应的, $W = E_{H_4(\sigma)}(M)$ 。在模块 628c 中,生成加密文本 $C = [U_0, U_2, \dots, U_{n+1}, V, W]$ 。该密文 C 中包括元素 $U_i = rP_{zi}$, 其中 $i = 0$ 和 $2 \leq i \leq n+1$, 其与接收方在分级结构中的位置有关。加密文本 C 的第二个组成部分是加密形式的随机二进制串 σ , $V = \sigma \oplus H_2(g^r)$, 其中 $g = \hat{e}(Q_0, P_{z1})$ 。元素 g 最好是第二循环群 Γ_2 的成员。加密文本 C 的第三个组成部分是 W , 如上所述,它是对称加密形式的实际消息。

[0101] 解密算法从模块 630a 开始,该模块示出了随机二进制串 σ 的恢复。随机二进制

串 σ 是利用公式 $s = V \oplus H_2\left(\frac{\hat{e}(U_0, S_{n+1})}{\prod_{i=2}^{n+1} \hat{e}(Q_{i-1}, U_i)}\right)$ 恢复的。然后利用公式 $M = E_{H_4(\sigma)}^{-1}(W)$ 从密

文 C 中恢复出消息 M (模块 630b)。可对加密文本进行检查,以检验内部一致性。例如,可以生成一个实验性的随机整数 $r' = H_3(\sigma, M, W)$, 如模块 630c 中所示。然后,就可以在模块 630d 中利用该实验性的随机整数 r' 来检验 $U_0 = r' P_0$ 及 $U_i = r' P_{zi}$, 其中 $2 \leq i \leq n+1$ 。如果成立,则可认为加密文本 C 是真实的。

[0102] 双基本 HIDE 与双全 HIDE

[0103] 参照图 2 和图 3 所述的双 HIDE 概念可被应用于基本 HIDE 和全 HIDE 方案。当发送方与接收方都处在分级结构中时,如图 3 所示,双 HIDE 就能令它们提高它们的加密通信的效率与安全性。对基本 HIDE 和全 HIDE 方案应用双 HIDE 需要决定附加信息,这些信息大多是通过上述的低级设置算法决定的。例如,必须为发送方的 m 个前辈低级 PKG 决定公共元素 P_{yi} 、低级密钥生成密文 s_{yi} 、低级机密元素 S_{yi} 以及低级密钥生成参数 Q_{yi} 。但是要注意,对于作为发送方 y 和接收方 z 的公共前辈的低级 PKG 来说,这些参数最好相同,以便对发送方 y 和接收方 z 进行分析(也就是说,对于所有的 $i \leq 1$ 最好有: $P_{yi} = P_{zi}, s_{yi} = s_{zi}, S_{yi} = S_{zi}$ 以及 $Q_{yi} = Q_{zi}$)。双 HIDE 还需要为发送方决定一个发送方公共元素 $P_{y(m+1)}$ 和一个发送方机密元素 $S_{y(m+1)}$,使用的方法与上述为接收方决定这些参数时所用的方法相同。

[0104] 有了这些附加参数,就可以根据双 HIDE 原理对消息 M 进行编码以生成一个加密文本 C ,编码的过程要利用低级密钥生成参数 Q_{yi} (其中 $i \geq 1$) 以及发送方机密元素 $S_{y(m+1)}$,但是不会用到 $i < 1$ 的低级密钥生成参数 Q_{yi} 。类似地,对加密文本 C 解码以恢复消息 M 时要利用低级密钥生成参数 Q_{yi} (其中 $i \geq 1$) 以及接收方机密元素 $S_{z(n+1)}$,但是不会用到 $i < 1$ 的低级密钥生成参数 Q_{zi} 。

[0105] 例如,在基本 HIDE 方案中(图 4 与图 5),双 HIDE 的应用改变了消息 M 的编码以生成一个密文 $C = [U_0, U_{1+1}, \dots, U_{n+1}, V]$,其中对于 $i = 0$ 和 $1+1 \leq i \leq n+1$,有 $U_i = rP_{zi}$,其中

中 $V = M \oplus H_2(g'_{yl})$, 并且其中 $g'_{yl} = \frac{\hat{e}(P_0, S_{y(m+1)})}{\prod_{i=l+1}^{m+1} \hat{e}(Q_{y(i-1)}, P_{yi})} \cdot U_i$ 因子的计算与以前相同,但是只需计算其中的少部分。然而,双基本 HIDE 要求发送方 y 使用多于上述生成 g 所必需的密钥生成参数 Q_{yi} 来生成 g_{yl} 。这是因为发送方的身份被包含到加密算法中了。解密算法的效率

提高更为惊人。消息 M 是利用 $M = V \oplus H_2\left(\frac{\hat{e}(U_0, S_{z(n+1)})}{\prod_{i=l+1}^{n+1} \hat{e}(Q_{z(i-1)}, U_{zi})}\right)$ 恢复的。同样,还是只需要较少的 U_i 参数。类似地,接收方需要用于双 HIDE 的密钥生成参数 Q_{zi} 比其他情况下所必需的要少。

[0106] 全 HIDE 也可以被修改以创建一种双全 HIDE 方案。加密算法中加密文本 C 的生成被修改为 $C = [u_0, U_{1+1}, \dots, U_{n+1}, V, W]$,其中对于 $i = 0$ 和 $1+1 \leq i \leq n+1$,有 $U_i = rP_{zi}$ 。参数 W 和 r 仍然以同样的方法生成, $W = E_{H_4(s)}(M)$, 并且 $V = \sigma \oplus H_2(g'_{yl})$ 中的参数

$$g'_{yl} = \frac{\hat{e}(P_0, S_{y(m+1)})}{\prod_{i=l+1}^{m+1} \hat{e}(Q_{y(i-1)}, P_{yi})} \cdot$$

[0107] 在双全 HIDE 方案中,解密算法也被修改。随机二进制串 σ 是利用

$$\sigma = V \oplus H_2\left(\frac{\hat{e}(U_0, S_{z(n+1)})}{\prod_{i=l+1}^{n+1} \hat{e}(Q_{z(i-1)}, U_{zi})}\right) \text{ 恢复的。另外,消息 } M \text{ 的恢复不变。}$$

[0108] 尽管已经用 $PKG_1 304b$ 作为发送方 y 与接收方 z 的最低公共前辈 PKG 说明了这些双 HIDE 方案,但是 $PKG_1 304b$ 可以是任意的公共前辈 PKG。加密与解密算法是相同的。但是为了获得最高的效率, $PKG_1 304b$ 最好是最低公共前辈 PKG。

[0109] 除了效率的提高以外,本发明的双 HIDE 方案还通过限制密钥托管提供了更高的安全性。在上述的基本 HIDE 和全 HIDE 方案中,接收方的所有直系前辈 PKG 都能解密发给

接收方的消息。但是,由于双 HIDE 方案加入了 PKG_{l-1} (PKG_l 的直接上代) 的密钥生成密文,该密文对于 PKG_{l-1} 以上的公共前辈 PKG 是未知的,因此那些公共前辈 PKG 就不能解密发送方 y 与接收方 z 之间的消息。

[0110] 密钥托管可被进一步限制,从而即使是 PKG_l 的直接上代也不能解密发送方 y 与接收方 z 之间的消息。这一点可以通过在为发送方 y 和接收方 z 生成私有密钥 (或是为 PKG_l 的后代同时又是发送方 y 与接收方 z 的前辈的 PKG 生成私有密钥) 的过程中隐藏 PKG_l 的私有密钥来实现。例如,对于某些随机的 $b \in \mathbb{Z}/q\mathbb{Z}$, PKG_l 304b 可以通过设置 $S'_{l-1} := S_l + bP_l$ 和 $Q'_{l-1} := Q_{l-1} + bP_0$ 而轻易地改变其私有密钥。新的私有密钥 S'_{l-1} 同样有效,但是却不为 PKG_l 的直接上代所知。因此, PKG_l 以上的 PKG 都不能对加密发送给接收方 z 的消息进行解码。更具体地说,只有接收方 z 在 PKG_l 的域内的前辈才能够解密发送给接收方 z 的消息。

[0111] 当 PKG_l 304b 通过设置 $S'_{l-1} := S_l + bP_l$ 和 $Q'_{l-1} := Q_{l-1} + bP_0$ 来改变其私有密钥时,新的私有密钥仍然与 PKG_{l-1} 的密钥生成密文 s_{l-1} 相关,因为新的私有密钥是从一个由 PKG_{l-1} 利用 s_{l-1} 生成的私有密钥推得的。一般而言,在本文所讨论的所有方案中,一个用户或 PKG 可以通过选取 b_i 的值 (其中 $1 \leq i \leq n$) 并设置 $S'_{z(n+1)} := S_{z(n+1)} + \sum_{i=1}^n b_i P_{z(i+1)}$ 和 $Q'_{zi} := Q_{zi} + b_i P_0$ (其中 $1 \leq i \leq n$), 来改变它自己的机密元素 $S_{z(n+1)}$ 和密钥生成参数 Q_{zi} (其中 $1 \leq i \leq n$)。然而,为了本发明的目的,这种新的私有密钥仍然被认为与原始的私有密钥有关,从而也和密钥生成密文 s_{zi} 的初始值有关。

[0112] 具有更高效的加密解密技术的双 HIDE 方案

[0113] 在上述的双 HIDE 方案中,可以将加密器必须计算的配对值的数量减少一个,而又不会提高解密器必须计算的配对值的数量。例如,上述的双基本 HIDE 加密算法可以被修改,从而密文 $C = [P_0, r(P_{y(l+1)} - P_{z(l+1)}), rP_{z(l+2)}, K, rP_{z(n+1)}, M \oplus H_2(g'_{y(l+1)})]$, 其中

$$g_{y(l+1)} = \frac{\hat{e}(P_0, S_{y(n+1)})}{\prod_{i=l+2}^m \hat{e}(Q_{y(i-1)}, P_{yi})} = \hat{e}(P_0, S_{y(l+1)}) \cdot \text{如果密文被表示为}$$

$$[0114] \quad C = [U_0, U_{l+1}, K, U_{n+1}, V], \text{那么就可以利用 } M = V \oplus H_2 \left[\frac{\hat{e}(U_0, S_{z(n+1)}) \hat{e}(U_{l+1}, Q_{zl})}{\prod_{i=l+2}^n \hat{e}(Q_{z(i-1)}, U_i)} \right] \text{来}$$

对其进行解密。

[0115] 同样地,也可以将解密器必须计算的配对值的数量减少一个,而又不会提高加密器必须计算的值的数量。例如,双基本 HIDE 加密算法可以被修改,从而密文 $C = [P_0, rP_{y(l+2)}, K, rP_{y(n)}, M \oplus H_2(g'_{z(l+1)})]$, 其中

$$g_{z(l+1)} = \frac{\hat{e}(P_0, S_{y(m+1)}) \hat{e}(Q_{yl}, (P_{z(l+1)} - P_{y(l+1)}))}{\prod_{i=l+2}^m \hat{e}(Q_{y(i-1)}, P_{yi})} = \hat{e}(P_0, S_{z(l+1)}) \cdot \text{如果密文被表示为}$$

$$[0116] \quad C = [U_0, U_{l+2}, K, U_n, V], \text{那么就可以利用 } M = V \oplus H_2 \left[\frac{\hat{e}(U_0, S_{z(n+1)})}{\prod_{i=l+2}^n \hat{e}(Q_{z(i-1)}, U_i)} \right] \text{来对其进}$$

行解密。

[0117] 经过认证的低级根 PKG

[0118] 可以通过创建一个经过认证的低级根 PKG 将上述双 HIDE 方案的高效性扩展到位

于分级结构之外的消息发送方。要“认证”低级 PKG, 根 PKG 可以提出一个附加参数, 比如一个随机消息 M' 。然后, 低级 PKG“签署” M' , 生成签名 $\text{Sig} = S_{z1} + s_{z1}P_{M'}$, 其中 S_1 是低级 PKG 的私有密钥, s_1 则是其低级密钥生成密文。低级 PKG 还会公布对应于 $1 \leq i \leq t$ 的 Q_i 。

[0119] 利用经过认证的低级根 PKG, 位于分级结构之外的发送方无需计算接收方所有 n 个前辈 PKG 的公共元素 P_{zi} 即可向接收方 z 发送加密消息。发送方可以利用对应于低级认证根 PKG 的参数更高效地加密消息。具体地说, 发送方计算 $P_{zi} = H_1(\text{ID}_1, \dots, \text{ID}_{zi}) \in \Gamma_1$, 其中 $1+1 \leq i \leq n+1$ 。然后发送方选取一个随机的 $r \in Z/qZ$, 并生成

加密文本 $C = [P_0, rP_{z(l+1)}, \dots, rP_{z(n+1)}, M \oplus H_2(g'_{zi})]$, 其中 $g_{zi} = \frac{\hat{e}(P_0, \text{Sig})}{\hat{e}(s_{zi}P_0, P_{M'})} = \hat{e}(P_0, S_{zi})$ 。

假设接收到的密文 $C = [U_0, U_{1+1}, \dots, U_{n+1}, V]$, 则接收方可以解密该密文以恢复消息

$M = V \oplus H_2 \left[\frac{\hat{e}(U_0, S_{z(n+1)})}{\prod_{i=l+1}^{n+1} \hat{e}(Q_{z(i-1)}, U_i)} \right]$, 其中 $S_{z(n+1)}$ 是接收方的私有密钥。

[0120] 分布式 PKG

[0121] 为了进一步保护上述 HIDE 方案的密钥生成密文, 并使得这些方案对于不诚实的 PKG 具有鲁棒性, 可以利用已知的阈加密技术将密钥生成密文和私有密钥分布开。

[0122] 更高效的加密技术

[0123] 通过将分级结构中的最高两级合并成单个根 PKG, 就可以提高用于上述 HIDE 方案的加密技术的效率。在那种情况下, $g = \hat{e}(Q_0, P_1)$ 被包含在系统参数中。这样就省去了加密器计算该配对值的任务。然而, 解密器必须计算一个额外的配对 (这是它在树的下一级的结果)。

[0124] HIDS 方案

[0125] 现在考虑本发明的签名方案或 HIDS 方案, 图 7 所示的流程图根据本发明的另一个当前优选实施例展示了一种生成及验证一个数字签名的方法。该方法在一个包含多个 PKG 的 HIDS 系统中实现。所述的 PKG 中至少包括一个根 PKG 以及根 PKG 与发送方或签署人之间的分级结构中的 n 个低级 PKG, 其中 $n \geq 1$ 。在模块 702 中, 根 PKG 选取一个只有根 PKG 知道的根密钥生成密文。该根密钥生成密文可被用来为分级结构中低于根 PKG 的 PKG 或用户产生私有密钥。然后在模块 704 中, 根 PKG 根据根密钥生成密文产生一个根密钥生成参数。在模块 706 中, 低级 PKG 选取低级密钥生成密文。与一个给定的低级 PKG 相关的低级密钥生成密文可被用来为分级结构中低于该相关低级 PKG 的 PKG 或用户生成私有密钥。与根密钥生成密文类似, 每个低级密钥生成密文只对其相关的低级 PKG 已知。在模块 708 中, 为 n 个低级 PKG 各自产生低级密钥生成参数。每个低级密钥生成参数的产生都至少要用到其相关低级 PKG 的低级密钥生成密文。

[0126] 在模块 710 中, 一个低级 PKG 为接收方生成一个私有密钥, 使得该私有密钥至少与 n 个低级密钥生成密文中的一个相关。例如, 发送方的私有密钥至少可以与向接收方发放私有密钥的 PKG 的低级密钥生成密文相关。但是, 接收方的私有密钥最好与它所有 n 个前辈 PKG 的低级密钥生成密文以及根密钥生成密文相关。在模块 712 中, 发送方至少利用其私有密钥来签署消息并生成数字签名。然后在模块 714 中, 接收方或校验器至少利用低级密钥生成参数之一来验证该数字签名。例如, 可以只利用根密钥生成参数来验证该签名。或者,

也可以利用低级密钥生成参数中的一个或多个。

[0127] 图 8 所示的流程图根据本发明的另一个当前优选实施例展示了一种生成及验证一条数字消息 M 的数字签名 Sig 的方法, 所述的数字消息在发送方 y 和接收方 z 之间传递。如图 3 所示, 发送方 y 306 在分级结构中比根 PKG 低 $m+1$ 个级别, 并且与 ID 元组 $(ID_{y1}, \dots, ID_{y(m+1)})$ 相关联。发送方的 ID 元组中包括与发送方相关的身份信息 $ID_{y(m+1)}$, 以及与它在分级结构中的 m 个前辈低级 PKG 的每一个相关的身份信息 ID_{yi} 。该方法从模块 802 开始, 生成元素的第一与第二循环群 Γ_1 和 Γ_2 。在模块 804 中, 选取一个函数 $\hat{e}$, 使得该函数 $\hat{e}$ 能够由第一循环群 Γ_1 的两个元素生成第二循环群 Γ_2 的一个元素。函数 $\hat{e}$ 最好是一种可行的配对, 如上所述。在模块 806 中选取第一循环群 Γ_1 的一个根生成器 P_0 。在模块 808 中, 选取一个随机根密钥生成密文 s_0 , 该密文与根 PKG 302 相关且只有根 PKG 302 知道。 s_0 最好是循环群 Z/qZ 的一个元素。在模块 810 中产生一个根密钥生成参数 $Q_0 = s_0 P_0$ 。 Q_0 最好是第一循环群 Γ_1 的一个元素。在模块 812 中, 选取一个第一函数 H_1 , 使得 H_1 能够由第一串二进制数生成第一循环群 Γ_1 的一个元素。在模块 814 中选取一个第二函数 H_3 , 使得 H_3 能够由第二循环群 Γ_2 的一个元素生成第二串二进制数。模块 802 至 814 的功能都是上述 HIDS 根设置算法的组成部分, 并且最好都在大致相同的时刻完成。作为示例, 比如那些在 Boneh-Franklin 中公开的函数就可以被用作 H_1 和 H_3 。实际上, 函数 H_1 和 H_3 可以是完全相同的函数。但是会有一个潜在的隐患。一名攻击者可能尝试让签署人签署 $M = ID_t$, 其中 ID_t 代表了一个实际身份。在这种情况下, 签署人的签名实际上会是一个私有密钥, 此后该密钥可被用来解密消息及伪造签名。但是, 通过采用某些能够区分签名与私有密钥提取的手段——比如一个比特前缀或为 H_3 采用不同的函数, 该隐患是可以避免的。

[0128] 接下来的一系列模块 (模块 816 至 824) 示出了作为低级设置算法的组成部分而执行的功能。在模块 816 中, 为发送方的 m 个前辈低级 PKG 各生成一个公共元素 P_{yi} 。每个公共元素 $P_{yi} = H_1(ID_1, \dots, ID_{yi})$ 最好都是第一循环群 Γ_1 的一个元素, 其中 $1 \leq i \leq m$ 。尽管是以单个模块表示的, 但是所有公共元素 P_{yi} 的产生需要进行一段时间, 而非一次全部完成。

[0129] 为发送方的 m 个前辈低级 PKG 304a, b, d 各选取一个低级密钥生成密文 s_{yi} (模块 818)。该低级密钥生成密文 s_{yi} 最好是循环群 Z/qZ 的元素, 其中 $1 \leq i \leq m$, 并且每个低级密钥生成密文 s_{yi} 最好只有与它相关的低级 PKG 知道。同样, 尽管是以单个模块表示, 但是所有低级密钥生成密文 s_{yi} 的选取需要进行一段时间, 而非一次全部完成。

[0130] 为发送方的 m 个前辈低级 PKG 各生成一个低级机密元素 S_{yi} (模块 820)。每个低级机密元素 $S_{yi} = S_{y(i-1)} + s_{y(i-1)} P_{yi}$ 最好是第一循环群 Γ_1 的一个元素, 其中 $1 \leq i \leq m$ 。尽管与公共元素 P_{yi} 以及密文 s_{yi} 一样都是以单个模块表示的, 但是所有机密元素 S_{yi} 的产生也需要进行一段时间, 而非一次全部完成。为了这些重复的密钥生成过程的缘故, 可将 S_0 定为 Γ_1 的身份元素。

[0131] 还要为发送方的 m 个前辈低级 PKG 各产生一个低级密钥生成参数 Q_{yi} (模块 822)。每个密钥生成参数 $Q_{yi} = s_{yi} P_0$ 最好都是第一循环群 Γ_1 的元素, 其中 $1 \leq i \leq m$ 。同样, 尽管以单个模块表示, 但是所有密钥生成参数 Q_{yi} 的产生也需要进行一段时间, 而非一次全部完成。

[0132] 随后两个模块 (模块 824 和 826) 的功能是作为上述抽取算法的一部分而执行的。

在模块 824 中生成与发送方 y 相关的发送方公共元素 $P_{y(m+1)}$ 。该发送方公共元素, $P_{y(m+1)} = H_1(ID_{y1}, \dots, ID_{y(m+1)})$, 最好是第一循环群 Γ_1 的一个元素。然后在模块 826 中生成与发送方 y 有关的发送方机密元素 $S_{y(m+1)}$ 。该发送方机密元素, $S_{y(m+1)} = S_{ym} + s_{ym}P_{y(m+1)} = \sum_{i=1}^{m+1} s_{y(i-1)}P_{yi}$, 最好也是第一循环群 Γ_1 的一个元素。

[0133] 为方便起见, 第一函数 H_1 可被选为一种迭代函数, 从而可以按照例如 $H_1(P_{y(i-1)}, ID_{yi})$ 而非 $H_1(ID_1, \dots, ID_{yi})$ 来计算公共点 P_i 。

[0134] 图 8 中所示的最后两个模块 (模块 828 和 830) 代表了上述的签署与验证算法。在模块 828 中, 消息 M 被签署以生成一个数字签名 Sig 。该签署过程最好至少用到发送方机密元素 $S_{y(m+1)}$ 。然后在模块 830 中验证数字签名 Sig 。该验证过程最好至少用到根密钥生成参数 Q_0 以及低级密钥生成参数 Q_{yi} 。现在将参照图 9 来说明这些参数和元素在签署消息 M 以及验证数字签名 Sig 时的具体用法。

[0135] 图 9 所示的流程图根据本发明的另一个当前优选实施例展示了一种生成及验证一条数字消息 M 的数字签名 Sig 的方法, 所述的数字消息在发送方 y 和接收方 z 之间传递。在该方案中, 根设置、低级设置以及抽取算法都与图 8 中模块 802 至 826 所示的实施例相同。因此, 图 9 所示的流程图在模块 927a 中从选取一个发送方密钥生成密文 $s_{y(m+1)}$ 开始, 该密文只有发送方 y 知道。在模块 927b 中, 利用公式 $Q_{y(m+1)} = s_{y(m+1)}P_0$ 产生一个与发送方相关联的发送方密钥生成参数 $Q_{y(m+1)}$ 。然后, 签署算法从发送方在模块 928a 中生成一个消息元素 $P_M = H_3(ID_{y1}, \dots, ID_{y(m+1)}, M)$ 。该消息元素 P_M 最好是第一循环群 Γ_1 的一个成员。在模块 928b 中利用方程 $Sig = S_{y(m+1)} + s_{y(m+1)}P_M$ 生成数字签名 Sig 本身。接收方通过检验方程

$$\frac{\hat{e}(P_0, Sig)}{\hat{e}(Q_{y(m+1)}, P_M) \prod_{i=2}^{m+1} \hat{e}(Q_{y(i-1)}, P_{yi})} = \hat{e}(Q_0, P_1) \text{ 是否满足来验证数字签名 } Sig。$$

[0136] 本文参照本发明的优选实施例以及图示实例对本发明进行了详细的说明, 但是应该明白, 在本发明的思想与范围内可以实现各种变化与改进。

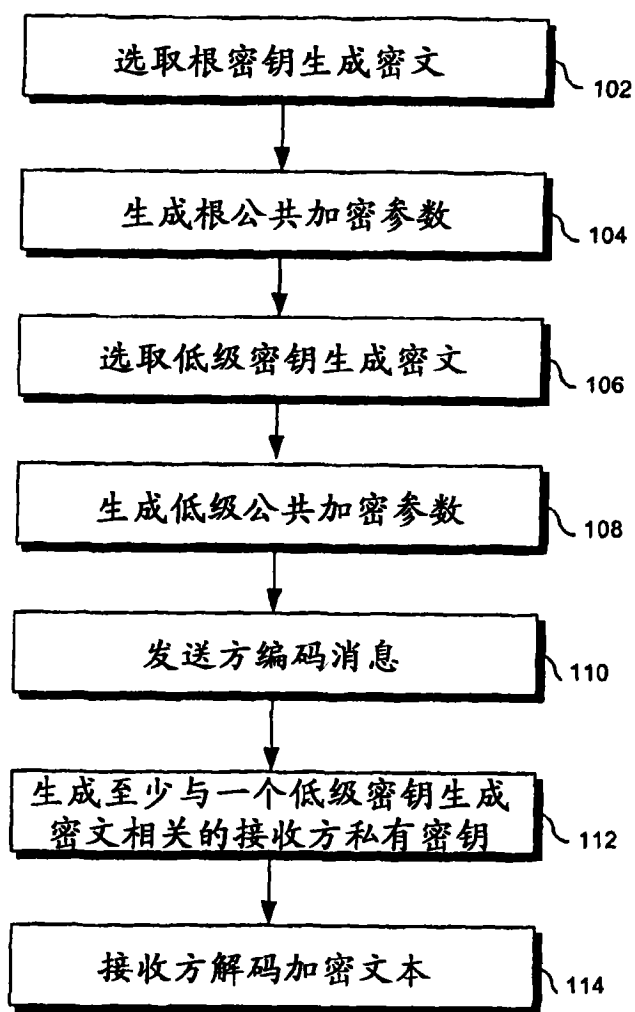


图 1

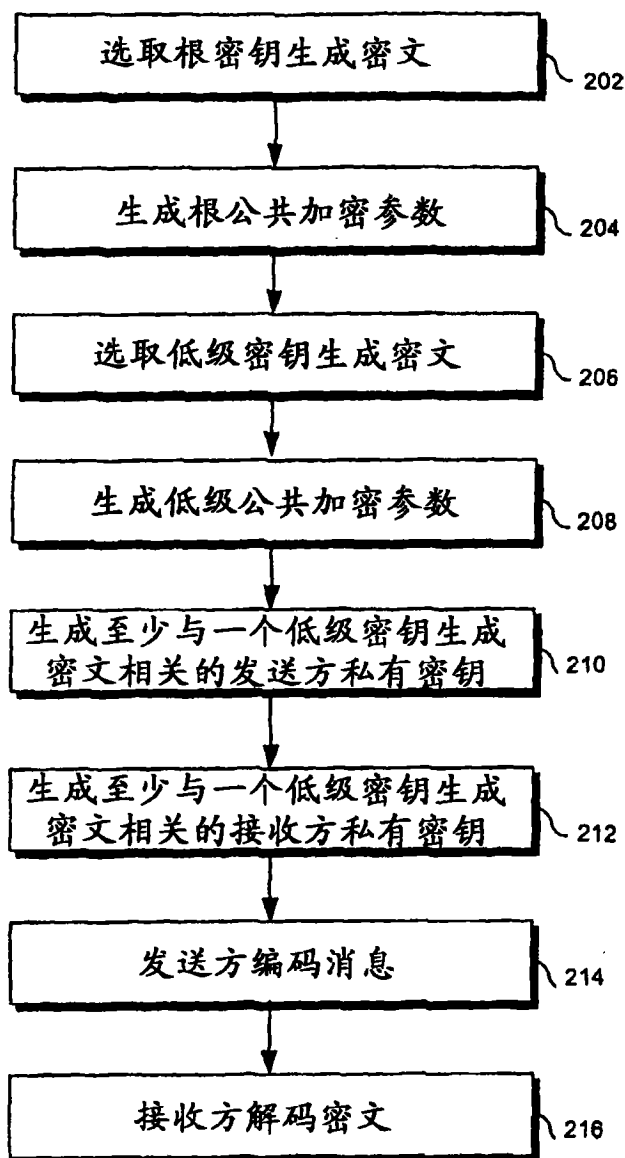


图 2

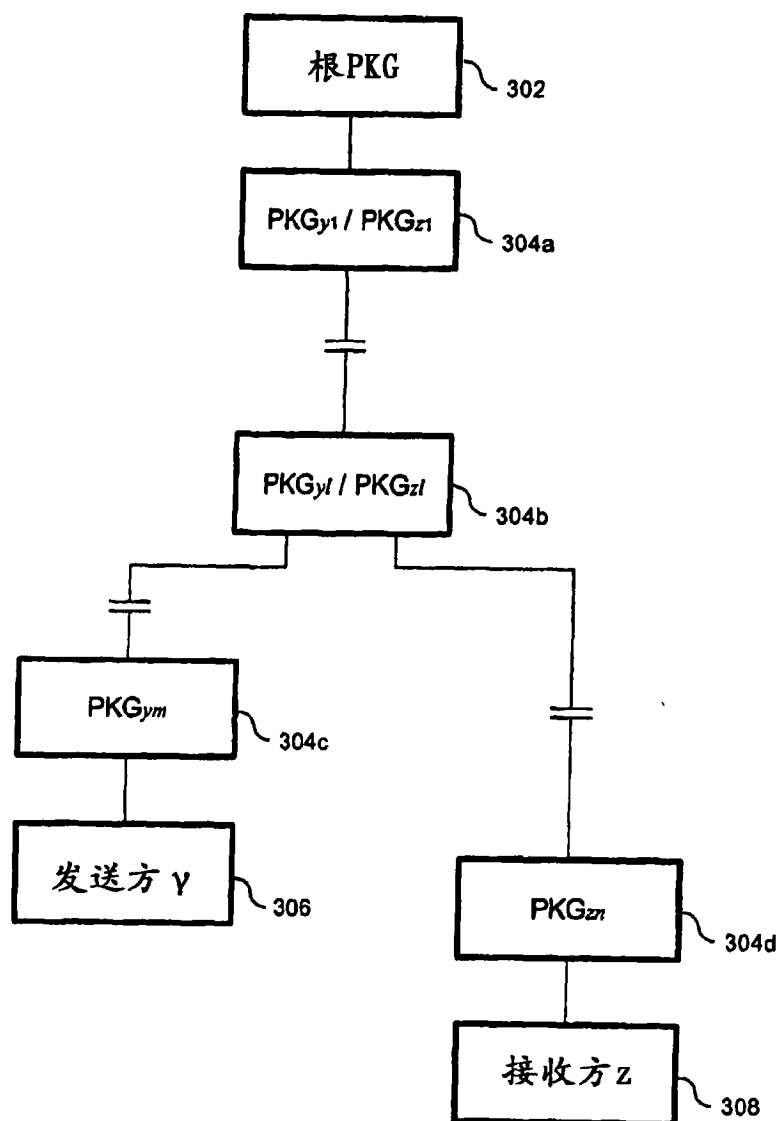


图 3

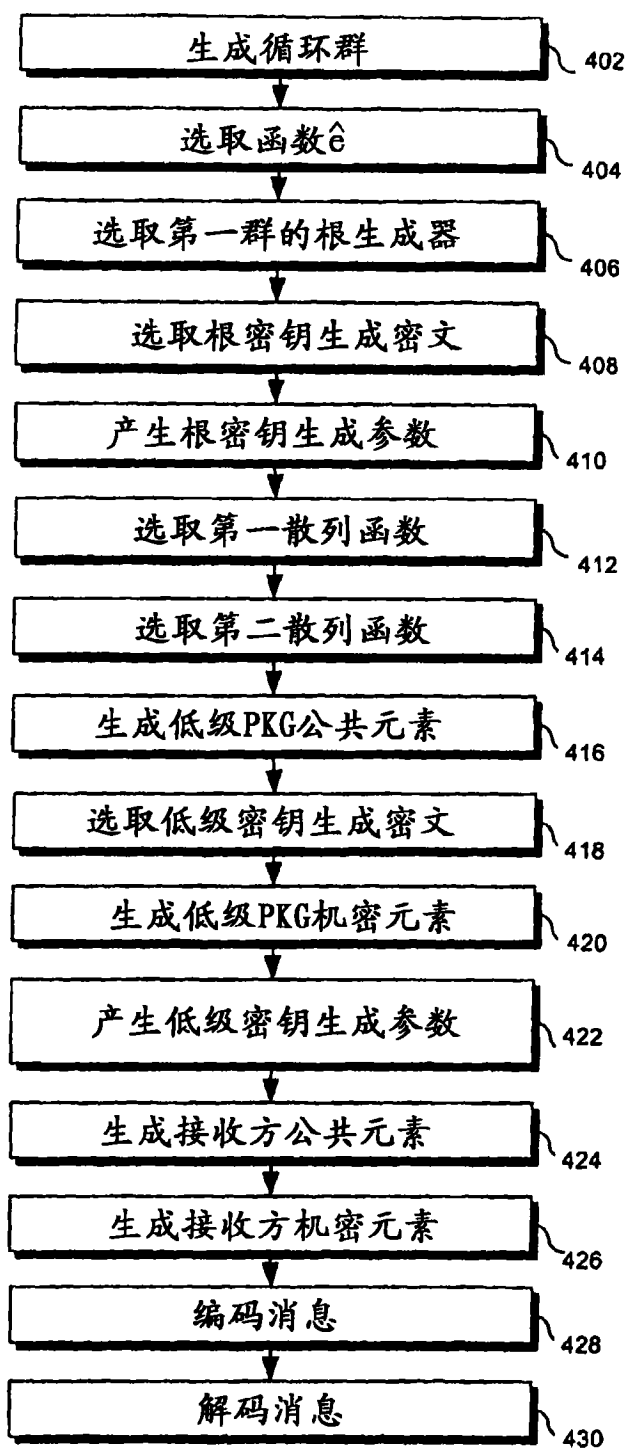


图 4

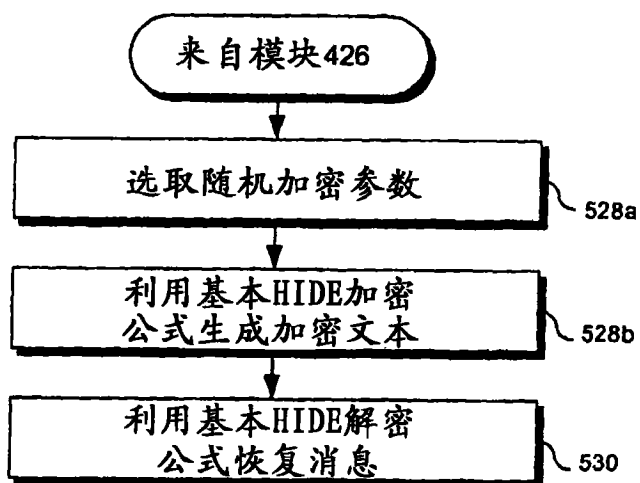


图 5

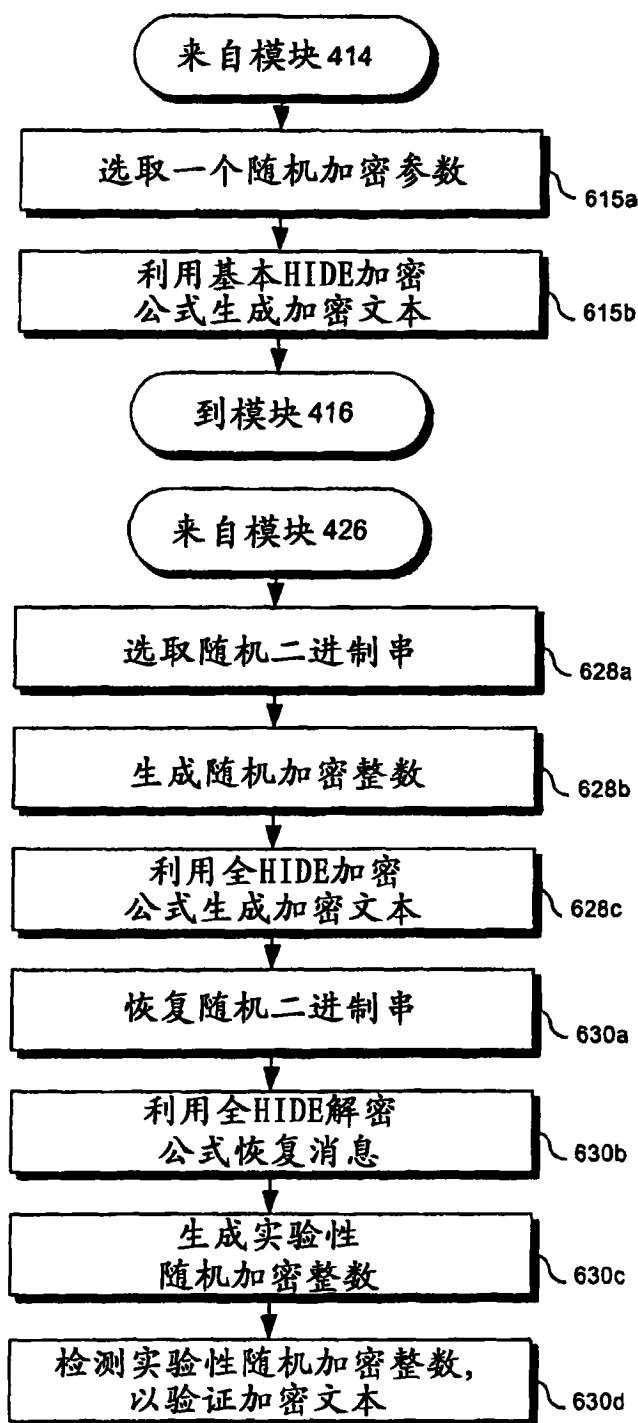


图6

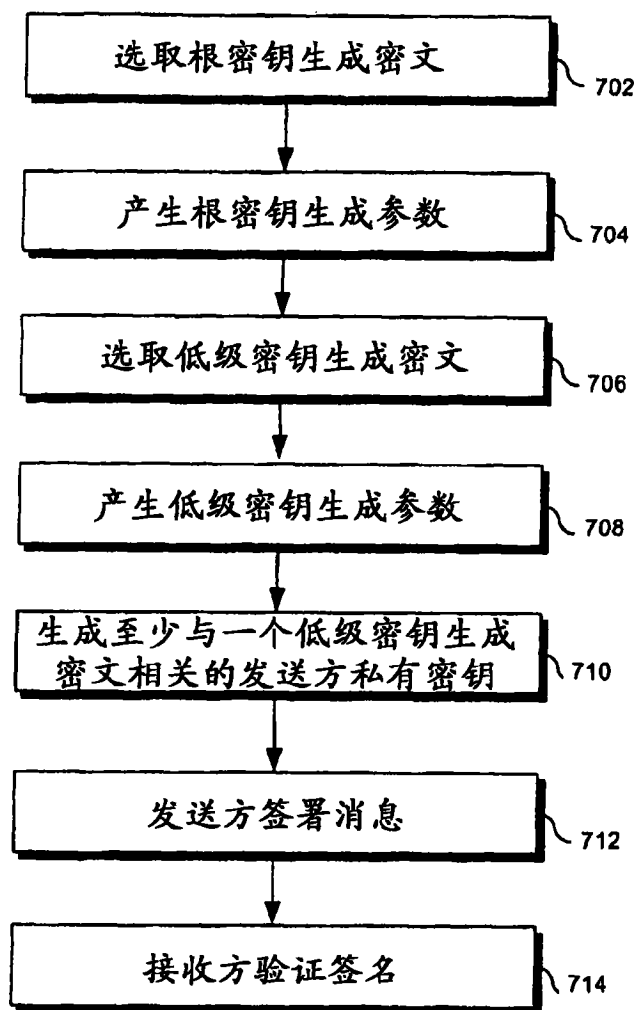


图 7

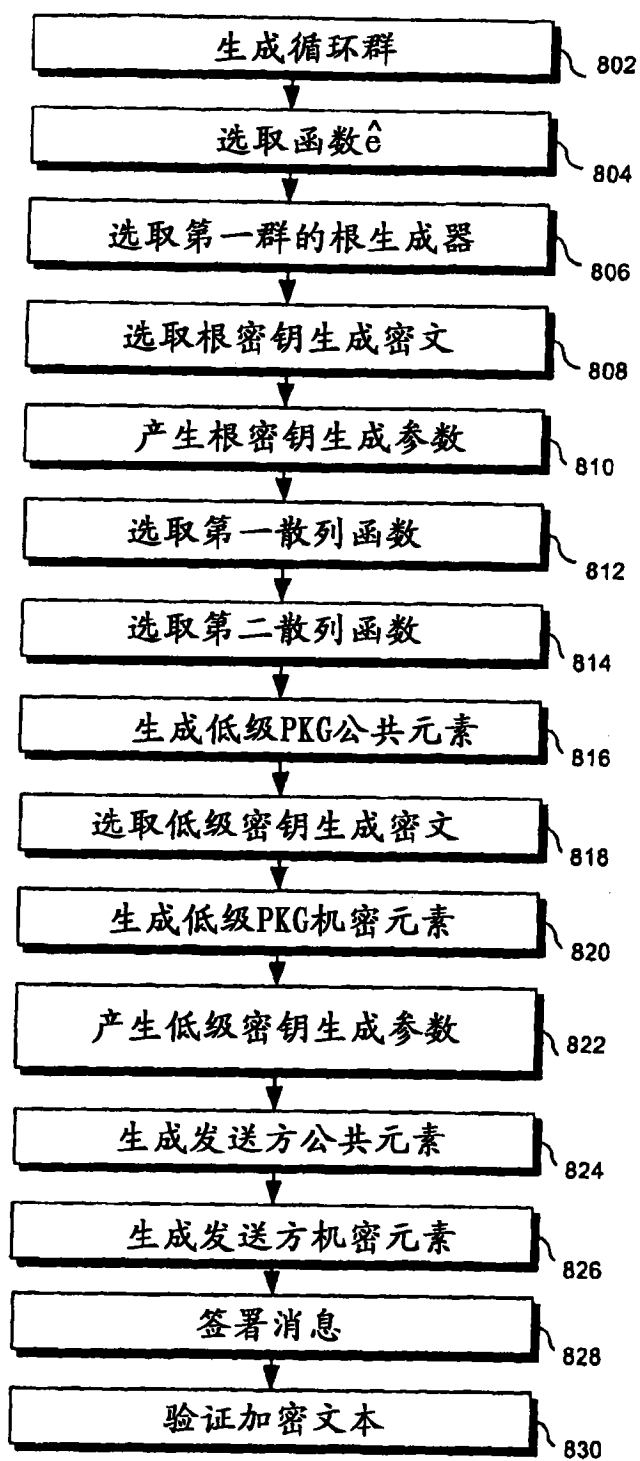


图 8

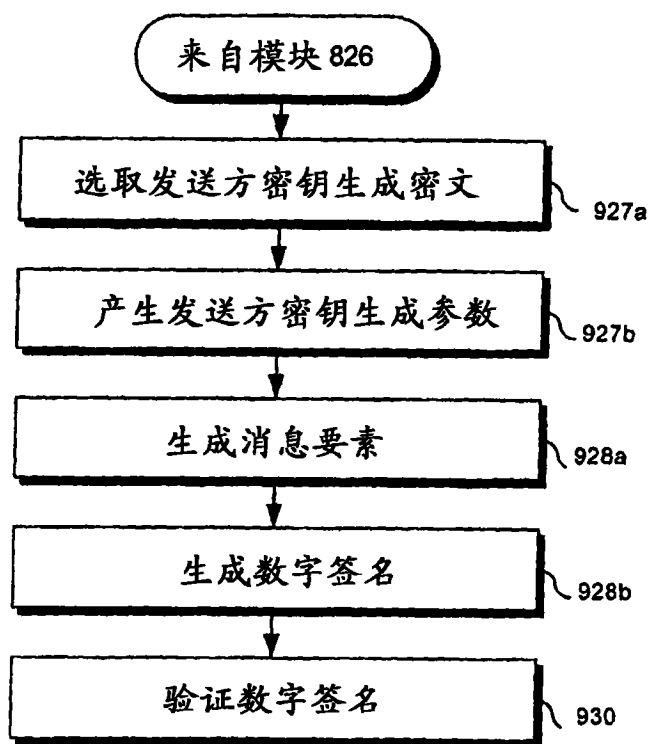


图 9