(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 104270249 B
(45)授权公告日 2017.10.17

(21)申请号 201410495127.5

(22)申请日 2014.09.23

(65)同一申请的已公布的文献号
申请公布号 CN 104270249 A

(43)申请公布日 2015.01.07

(73)专利权人 电子科技大学
地址 611731 四川省成都市高新区(西区)西源大道2006号

(72)发明人 李发根 吴威峰

(74)专利代理机构 电子科技大学专利中心 51203
代理人 周刘英

(51)Int.Cl.
H04L 9/32(2006.01)

(56)对比文件
CN 103905189 A,2014.07.02,
CN 103746810 A,2014.04.23,
CN 102983971 A,2013.03.20,
CN 103297963 A,2013.09.11,
WO 2014071719 A1,2014.05.15,

审查员 周思

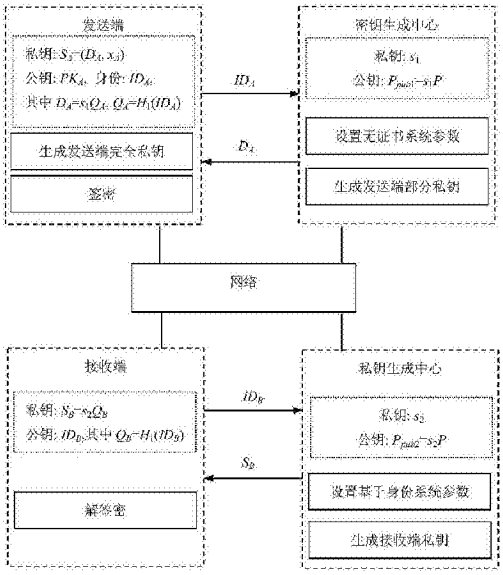
权利要求书1页 说明书6页 附图3页

(54)发明名称

一种从无证书环境到基于身份环境的签密方法

(57)摘要

本发明公开了一种从无证书环境到基于身份环境的签密方法。属于保密通信领域。为了使基于无证书环境的发送端能够利用签密方法发送消息给基于身份环境的接收端，本发明包括：初始化系统参数后，密钥生成中心生成部分私钥D_A并发送给发送端，发送端基于系统参数随机生成一个秘密值与部分私钥D_A设置完全私钥S_A和公钥PK_A；私钥生成中心生成私钥S_B并发送给接收端；发送端根据系统参数、发送端的身份与完全私钥S_A、公钥PK_A、接收端的身份和消息m，生成签密密文，并将该密文和发送端身份和公钥PK_A发送给接收端进行解签密处理。本发明用于异构网络的签密通信，为无证书环境的用户和基于身份环境的用户提供端到端的机密性、完整性、认证和不可否认性服务。



CN 104270249 B

1.一种从无证书环境到基于身份环境的签密方法，其特征在于，包括下列步骤：

设定系统参数，包括$\{G_1,G_2,p,\hat{e},H_1,H_2,H_3,H_4,t\}$，其中$G_1$表示阶为p的循环加法群，$G_2$表示阶为p的循环乘法群，$\hat{e}$表示$G_1\times G_1\rightarrow G_2$的双线映射，$H_1$～$H_4$表示哈希函数，其中$H_1$、$H_3$和$H_4$为从$\{0,1\}^*$映射到$G_1^*$，$H_2$从$\{0,1\}^*$映射到$\{0,1\}^t$，其中$\{0,1\}^*$表示任意比特长的二进制序列组成的集合，$G_1^*$表示去掉单位元所得到的加法群，$\{0,1\}^t$表示比特长度为t的二进制序列组成的集合，其中t为预设参数，Z_p^*表示有限域$Z_p=\{0,1,\cdots,p-1\}$去掉元素零所得到集合；

无证书环境的发送端提交身份信息ID_A给密钥生成中心，密钥生成中心根据系统参数和身份信息ID_A生成部分私钥D_A并发送给发送端，其中$D_A=s_1Q_A$，其中$s_1\in Z_p^*$，$Q_A=H_1(ID_A)$；

发送端基于系统参数随机生成一个秘密值x_A，根据秘密值x_A与部分私钥D_A设置完全私钥S_A和公钥PK_A，其中秘密值$x_A\in Z_p^*$，公钥$PK_A=x_AP$，P表示循环加法群G_1的生成元；

基于身份环境的接收端提交身份信息ID_B给私钥生成中心，私钥生成中心根据系统参数和身份信息ID_B生成私钥S_B并发送给接收端，其中$S_B=s_2Q_B$，其中$s_2\in Z_p^*$，$Q_B=H_1(ID_B)$；

发送端根据系统参数、发送端的身份信息ID_A、发送端的公钥PK_A和完全私钥S_A、接收端的身份信息ID_B以及消息m，生成签密密文σ，并将签密密文σ和公钥PK_A、身份信息ID_A发送给接收端；其中生成签密密文σ具体为：随机选择有限域Z_p^*中的任一元素r，计算承诺$V=rP$和$T=\hat{e}(P_{pub2},Q_B)^r$，其中公钥$P_{pub2}=s_2P$；基于哈希函数$H_2$计算哈希值$h=H_2(V,T,ID_B)$，从而生成关于消息m的签密密文$\sigma=(V,c,W)$，其中$c=m\oplus h$，签名$W=D_A+rH_3(V,c,ID_A,PK_A)+x_AH_4(V,c,ID_A,PK_A)$；

接收端根据系统参数、发送端的身份信息ID_A和公钥PK_A、接收端的身份信息ID_B和私钥S_B，对密文σ进行解签密处理：判断$\hat{e}(W,P)=\hat{e}(P_{pub1},Q_A)\hat{e}(H_3(V,c,ID_A,PK_A),V)\hat{e}(H_4(V,c,ID_A,PK_A),PK_A)$是否成立，若否，则认为当前密文σ无效，其中公钥$P_{pub1}=s_1P$；否则基于私钥$S_B$恢复承诺$T=\hat{e}(V,S_B)$后，由哈希函数$H_2$计算哈希值$h=H_2(V,T,ID_B)$，基于$m=c\oplus h$输出消息m。

2.如权利要求1所述的方法，其特征在于，所述完全私钥$S_A=x_A||D_A$，其中符号“||”表示比特级联。

3.如权利要求1所述的方法，其特征在于，所述公钥$P_{pub1}=s_1P_1$，$P_{pub2}=s_2P_2$，其中P_1和P_2表示循环加法群G_1的不同生成元；

在生成签密密文$\sigma=(V_1,V_2,c,W)$时，哈希值$h=H_2(V_1,V_2,T,ID_B)$，其中$V_1=rP_1$、$V_2=rP_2$，签名$W=D_A+rH_3(V_1,V_2,c,ID_A,PK_A)+x_AH_4(V_1,V_2,c,ID_A,PK_A)$；

解签密处理时，判断$\hat{e}(W,P_1)=\hat{e}(P_{pub1},Q_A)\hat{e}(H_3(V_1,V_2,c,ID_A,PK_A),V_1)\hat{e}(H_4(V_1,V_2,c,ID_A,PK_A),PK_A)$是否成立，若是，则根据$T=\hat{e}(V,S_B)$恢复承诺T后，计算$h=H_2(V_1,V_2,T,ID_B)$。

4.如权利要求2或3所述的方法，其特征在于，预设参数t为消息m的比特长度。

一种从 无证书环境到基于身份环境的签密方法

技术领域

[0001] 本发明属于保密通信技术领域，特别是涉及一种发送端属于无证书环境、接收端属于基于身份环境的异构签密方法。

背景技术

[0002] 密码体制是实现保密通信的重要工具。密码体制提供的基本安全服务有机密性(confidentiality)、完整性(integrity)、认证(authentication)和不可否认性(non-repudiation)。机密性是指信息只为授权用户使用，不能泄露给未授权的用户。完整性是指信息在传输或存储过程中，不能被偶然或蓄意地删除、修改、伪造、重放、插入等破坏和丢失的特性。认证是确保通信方的确是它所声称的那位。确认一个实体的身份称为实体认证，确认一个信息的来源称为消息认证。不可否认性是防止通信方对以前的许诺或者行为的否认。在密码体制中，机密性可以通过一种基本的密码原语称为加密(encryption)来取得。加密可以看成是一种变换，这种变换将可读的明文信息变换成不可读的密文信息。数字签名(digital signature)也是一种基本的密码原语，它可以取得完整性、认证和不可否认性。数字签名可以看成是对数据所做的一种密码变换，这种密码变换可以使数据的接收端确认签名者的身份和数据的完整性。如果我们需要同时取得机密性、完整性、认证和不可否认性，一个传统的方法是先对消息进行签名，然后再进行加密，称为“先签名后加密”方法。这种方法的计算量和通信成本是加密和签名代价之和，效率较低。1997年，Zheng提出了一种新的密码原语来同时取得这四种安全性质，他称这一密码原语为数字签密。比起传统的“先签名后加密”方法，签密具有以下优点：

[0003] (1)签密在计算量和通信成本上都要低于传统的“先签名后加密”方法。

[0004] (2)签密允许并行计算一些昂贵的密码操作。

[0005] (3)合理设计的签密可以取得更高的安全水平。

[0006] (4)签密可以简化同时需要保密和认证的密码协议的设计。

[0007] 1976年，Diffie和Hellman提出了公钥密码体制的概念，解决了对称密码体制中最难解决的两个问题：密钥分配和数字签名。在公钥密码体制中，每个用户拥有两个密钥：私钥和公钥，其中只有私钥由用户秘密保存，公钥可以由一个证书权威(certificate authority,CA)保存在一个公钥目录中。然而，公钥密码体制易受到“公钥替换”攻击，即攻击者用自己选定的假公钥替换一个公钥目录中真实的公钥。当一个用户用这个假公钥加密一个消息时，这个攻击者就可以正确地解密。为了抵抗公钥替换攻击，需要让用户的公钥以一种可验证和可信的方式与用户的身份信息关联起来。目前，认证用户的公钥有三种方法：基于公钥基础设施(public key infrastructure,PKI)的方法、基于身份(identity-based)的方法和无证书(certificateless)方法。事实上，可以根据公钥认证方法的不同，把公钥密码体制分为基于公钥基础设施的密码体制、基于身份的密码体制和无证书密码体制。下面解释这三种密码体制的特点。

[0008] (1)基于公钥基础设施的密码体制：每个用户的公钥都伴随一个公钥证书，这个公

钥证书由CA签发。公钥证书是一个结构化的数据记录，它包括了用户的身份信息、公钥参数和 CA的签名等。任何人都可以通过验证证书的合法性(CA的签名)来认证公钥。这种方法有如下两个缺点：①使用任何公钥前都需要先验证公钥证书的合法性，增加了用户的计算量；②CA需要管理大量的证书，包括证书的颁发、存储、撤销等。

[0009] (2)基于身份的密码体制：为了简化密钥管理，Shamir于1984年首次提出了基于身份的密码体制的概念[Shamir A.Identity-based cryptosystems and signature schemes.Advances in Cryptology-CRYPTO'84,LNCS 196,1985:47-53.]。在基于身份的密码体制中，用户的公钥可以根据用户的身份信息(如姓名、身份证号码、电话号码、E-mail地址等)直接计算出来，用户的私钥则是由一个称为私钥生成中心(private key generator,PKG)的可信方生成。基于身份的密码体制取消了公钥证书，减少了公钥证书的存储和合法性验证。但是，基于身份的密码体制有一个致命的缺点：所有用户的私钥都由PKG生成。PKG知道所有用户的私钥不可避免的引起密钥托管问题。

[0010] (3)无证书密码体制：为了克服基于身份的密码体制中的密钥托管问题，Al-Riyami 和Paterson提出了无证书密码体制(certificateless cryptography)的概念[Al-Riyami S S,Paterson K G.Certificateless public key cryptography.Advances in Cryptology-ASIACRYPT 2003,LNCS 2894,2003:452-473.]。在这种密码体制中，用户的私钥来自于两部分，一部分是用户自己选择的秘密值，一部分是由密钥生成中心(key generating centre,KGC)根据用户的身份信息计算的部分私钥。公钥通常利用秘密值来生成，但这里的公钥不必有单独认证的公钥证书。也就是说，用户需要联合KGC生成的部分私钥和自己的秘密值来生成完全私钥。KGC并不知道用户的完全私钥，从而消除了密钥托管问题。

[0011] 在过去的研究过程中，人们通常都假设参与方属于相同的公钥认证环境，即两方或者多方要么同属于公钥基础设施环境、要么同属于在基于身份环境、要么同属于无证书环境。然而，现代社会形成的互联全球的计算机和通信系统是非常不同类的。物联网、云计算这些新技术的出现加重了网络的异构程度。不同的国家、地区和企业可能采用不同的网络技术和不同的安全技术(这里主要指公钥认证技术的不同)。

[0012] 当前，在针对网络异构的签密方案主要有以下几种方案：

[0013] 2010年，Sun和Li提出的一方属于公钥基础设施环境、另一方属于基于身份环境的异构签密方案[Sun Y,Li H.Efficient signcryption between TPKC and IDPKC and its multi-receiver construction.SCIENCE CHINA Information Sciences,2010,53(3):557-566.]，但是该方案只满足外部安全性(即攻击者不能是发送端或者接收端)，这样的方案不能提供否认性。同时该方案还存在内部安全性问题，即如果发送端的私钥丢失了，攻击者也不能从密文中恢复出消息；如果接收端的私钥丢失了，攻击者也不能伪造一个密文。

[0014] 2011年，Huang,Wong和Yang提出了两个发送端属于公钥基础设施环境、接收端属于基于身份环境的异构签密方案[Huang Q,Wong D S,Yang G.Heterogeneous signcryption with key privacy.The Computer Journal,2011,54(4):525-536.]，该方案满足内部安全性。

[0015] 2013年，李发根、张辉和Takagi提出了两个异构签密方案[Li F,Zhang H,Takagi T.Efficient signcryption for heterogeneous systems.IEEE Systems Journal,2013,

7(3):420-429.],第一个方案允许属于公钥基础设施环境中的发送端发送消息给属于基于身份环境中的接收端,第二个方案允许属于基于身份环境中的发送端发送消息给属于公钥基础设施环境中的接收端,这两个方案都达到了内部安全性。

[0016] 2013年,李发根和熊盼将异构签密和在线/离线签名相结合,设计了一个发送端属于基于身份环境、接收端属于公钥基础设施环境的在线/离线签密方案[Li F,Xiong P.Practical secure communication for integrating wireless sensor networks into the Internet of things.IEEE Sensors Journal,2013,13(10):3677-3684.],该方案被应用于解决物联网中的安全问题,其发送端是一个传感器节点,接收端是一个Internet主机。为了降低传感器节点的计算成本,该方案将签密分为两个阶段:离线阶段和在线阶段。离线阶段在不知道消息的情况下完成大部分计算工作。当消息可用的时候,在线阶段只需要完成很少的计算就完成了整个签密过程。

[0017] 另外,中国专利申请CN103746811A公开了一种发送端属于基于身份环境、接收端属于公钥基础设施环境的匿名签密方法,中国专利申请103746810A公开了一种发送端属于公钥基础设施环境、接收端属于基于身份环境的匿名签密方法。

[0018] 但是,上述签密方法都不能适用发送端属于无证书环境、接收端属于基于身份环境的通信问题。

发明内容

[0019] 本发明的目的在于:实现从无证书环境到基于身份环境的签密通信,为无证书环境的用户和基于身份环境的用户提供端到端的安全保障。

[0020] 本发明公开了一种从无证书环境到基于身份环境的签密方法,包括:

[0021] 系统初始化:设定系统参数,用于生成发送端的私钥D_A、接收端私钥S_B,以及发送端的签密和接收端的解签密;

[0022] 无证书环境的发送端提交身份信息ID_A给密钥生成中心,密钥生成中心根据系统参数和身份信息ID_A生成部分私钥D_A并发送给发送端;发送端基于系统参数随机生成一个秘密值x_A,根据秘密值x_A与部分私钥D_A设置完全私钥S_A和公钥PK_A,其中私钥S_A可设置为$S_A=x_A||D_A$,其中符号“||”表示比特级联;

[0023] 基于身份环境的接收端提交身份信息ID_B给私钥生成中心,私钥生成中心根据系统参数和身份信息ID_B生成私钥S_B并发送给接收端;

[0024] 发送端根据系统参数、发送端的身份信息ID_A、发送端的公钥PK_A和完全私钥S_A、接收端的身份信息ID_B以及消息m,生成签密密文σ,并将签密密文σ和公钥PK_A、身份信息ID_A发送给接收端;

[0025] 接收端根据系统参数、发送端的身份信息ID_A和公钥PK_A、接收端的身份信息ID_B和私钥S_B,对密文σ进行解签密处理。

[0026] 由于采用了上述技术方案,本发明的有益效果是:为无证书环境的用户和基于身份环境的用户提供端到端的机密性、完整性、认证和不可否认性服务。

附图说明

[0027] 本发明将通过例子并参照附图的方式说明,其中:

[0028] 图1是本发明具体实施方式的签密操作流程图；

[0029] 图2是本发明具体实施方式的解签密操作流程图；

[0030] 图3是本发明的实施例1的系统结构示意；

[0031] 图4是本发明的实施例2的系统结构示意。

具体实施方式

[0032] 为使本发明的目的、技术方案和优点更加清楚，下面结合实施方式和附图，对本发明作进一步地详细描述。

[0033] 实施例1

[0034] 参见图3，具体执行步骤包括设定系统参数、生成无证书环境的密钥、生成基于身份环境的密钥、签密和解签密，具体描述如下：

[0035] (1)设定系统参数

[0036] (1.1)设G_1为由P生成的循环加法群，阶为p，G_2为具有相同阶p的循环乘法群，$\hat{e}:G_1\times G_1\to G_2$为一个双线性映射。定义四个安全Hash函数$H_1$，$H_2$，$H_3$和$H_4$。$H_1$、$H_3$和$H_4$都是从{0,1}*映射到$G_1^*$，$H_2$从{0,1}*映射到{0,1}^{t}，其中{0,1}*表示任意比特长的二进制序列组成的集合，G_1^*表示去掉单位元所得到的加法群，{0,1}^{t}表示比特长度为t(t为预设参数，本实施例中取值设定为消息m的比特长度l_m)的二进制序列组成的集合，Z_p^*表示有限域 $Z_p=\{0,1,\cdots,p-1\}$去掉元素零所得到的集合。

[0037] 基于上述设定，得到的系统参数为：$\{G_1,G_2,p,\hat{e},P,H_1,H_2,H_3,H_4,l_m\}$。

[0038] (1.2)密钥生成中心随机选择一个私钥$s_1\in Z_p^*$，计算相应的公钥$P_{pub1}=s_1P$。

[0039] (1.3)私钥生成中心随机选择一个私钥$s_2\in Z_p^*$，计算相应的公钥$P_{pub2}=s_2P$。

[0040] (2)生成无证书环境的密钥

[0041] 发送端提交身份信息ID_A给密钥生成中心，密钥生成中心计算部分私钥$D_A=s_1Q_A$，其中 $Q_A=H_1(ID_A)$，密钥生成中心将部分私钥D_A安全的发送给发送端。

[0042] (2.2)发送端随机选择$x_A\in Z_p^*$作为秘密值。

[0043] (2.3)发送端设置完全私钥$S_A=(D_A,x_A)$，即$S_A=x_A||D_A$。

[0044] (2.4)发送端计算公钥$PK_A=x_AP$。

[0045] (3)生成基于身份环境的密钥

[0046] (3.1)接收端提交身份信息ID_B给私钥生成中心，私钥生成中心计算私钥$S_B=s_2Q_B$，其中 $Q_B=H_1(ID_B)$，私钥生成中心将私钥S_B发送给接收端。

[0047] (4)签密

[0048] 发送端在获取到接收端的身份ID_B后，可以利用身份信息ID_A、公钥PK_A、完全私钥S_A对消息m进行签密。参见图1，具体步骤如下：

[0049] (4.1)随机选取$r\in Z_p^*$，计算承诺V、T：$V=rP$、$T=\hat{e}(P_{pub2},Q_B)^r$。

[0050] (4.2)根据哈希函数H_2，计算哈希值$h=H_2(V,T,ID_B)$。

[0051] (4.3)计算$c=m\oplus h$，符号“$\oplus$”表示异或运算。

[0052] (4.4)计算签名$W=D_A+rH_3(V,c,ID_A,PK_A)+x_AH_4(V,c,ID_A,PK_A)$。

[0053] (4.5)发送消息m的签密密文σ＝(V,c,W)和发送端的身份信息ID_A与公钥PK_A给接收端。

[0054] (5)解签密

[0055] 接收端在收到密文σ＝(V,c,W)和发送端身份信息ID_A与公钥PK_A时，参见图2，具体执行以下步骤：

[0056] (5.1)验证签名W的合法性

[0057] 检查等式$\hat{e}(W,P)=\hat{e}(P_{pub1},Q_A)\hat{e}(H_3(V,c,ID_A,PK_A),V)\hat{e}(H_4(V,c,ID_A,PK_A),PK_A)$是否成立，如果上式不成立，则认为当前密文无效，拒绝该密文；否则继续执行下面的步骤5.2。

[0058] (5.2)基于接收端的私钥S_B恢复承诺T，即计算$T=\hat{e}(V,S_B)$。

[0059] (5.3)根据哈希函数H_2，步骤(5.2)计算的T值，计算$h=H_2(V,T,ID_B)$。

[0060] (5.4)再根据步骤(5.3)计算的h值，恢复并输出消息$m=c\oplus h$。

[0061] 实施例2

[0062] 参见图4，具体执行步骤包括设定系统参数、生成无证书环境的密钥、生成基于身份环境的密钥、签密和解签密。实施例2和实施例1的主要区别在于无证书环境和基于身份环境选择的生成元不同。

[0063] (1)设定系统参数

[0064] (1.1)与实施例1的设定方式相同，设定系统参数$\{G_1,G_2,p,\hat{e},H_1,H_2,H_3,H_4,l_m\}$。

[0065] (1.2)密钥生成中心选择群G_1的一个生成元P_1和一个主私钥$s_1\in Z_p^*$，计算相应的公钥 $P_{pub1}=s_1P_1$。

[0066] (1.3)私钥生成中心选择群G_1的一个生成元P_2和一个主私钥$s_2\in Z_p^*$，计算相应的公钥 $P_{pub2}=s_2P_2$。

[0067] (2)生成无证书环境的密钥

[0068] 发送端提交身份信息ID_A给密钥生成中心，密钥生成中心计算部分私钥$D_A=s_1Q_A$，其中 $Q_A=H_1(ID_A)$，密钥生成中心将部分私钥D_A安全的发送给发送端。

[0069] (2.2)发送端随机选择$x_A\in Z_p^*$作为秘密值。

[0070] (2.3)发送端设置完全私钥$S_A=(D_A,x_A)$，即$S_A=x_A||D_A$。

[0071] (2.4)发送端计算公钥$PK_A=x_A\ P_1$。

[0072] (3)生成基于身份环境的密钥

[0073] (3.1)接收端提交身份信息ID_B给私钥生成中心，私钥生成中心计算私钥$S_B=s_2Q_B$，其中 $Q_B=H_1(ID_B)$，私钥生成中心将私钥S_B发送给接收端。

[0074] (4)签密

[0075] 发送端在获取到接收端的身份ID_B后，可以利用身份信息ID_A、公钥PK_A、完全私钥S_A对消息m进行签密。参见图1，具体步骤如下：

[0076] (4.1)随机选取$r\in Z_p^*$，计算承诺V_1、V_2、T：$V_1=rP_1$、$V_2=rP_1$、$T=\hat{e}(P_{pub2},Q_B)^r$。

[0077] (4.2)根据哈希函数H_2，计算哈希值$h=H_2(V_1,V_2,T,ID_B)$。

[0078] (4.3)计算$c=m\oplus h$。

[0079] (4.4)计算签名$W=D_A+rH_3(V_1,V_2,c,ID_A,PK_A)+x_AH_4(V_1,V_2,c,ID_A,PK_A)$。

[0080] (4.5)发送消息m的签密密文σ＝(V,c,W)和发送端的身份信息ID_A与公钥PK_A给接收

端。

[0081] (5)解签密

[0082] 接收端在收到密文σ＝(V,c,W)和发送端身份信息ID_A与公钥PK_A时，参见图2，具体执行以下步骤：

[0083] (5.1)检查等式

[0084] $\hat{e}(W,P_1)=\hat{e}(P_{pub1},Q_A)\hat{e}(H_3(V_1,V_2,c,ID_A,PK_A),V_1)\hat{e}(H_4(V_1,V_2,c,ID_A,PK_A),PK_A)$，

如果上式不成立，则认为当前密文无效，拒绝该密文；否则继续执行下面的步骤5.2。

[0085] (5.2)基于接收端的私钥S_B恢复承诺T，即计算$T=\hat{e}(V,S_B)$。

[0086] (5.3)根据哈希函数H_2，步骤(5.2)计算的T值，计算$h=H_2(V_1,V_2,T,ID_B)$，其中$V_1=rP_1$、$V_2=rP_1$。

[0087] (5.4)再根据步骤(5.3)计算的h值，恢复并输出消息$m=c\oplus h$。

[0088] 申请人在英特尔酷睿(Intel Core)i7 4770S处理器(3.10GHz)、内存为4G的计算机上，利用PBC库(选择类型A配对)实现了上述两种实施方式。对于第一种实施方式来说，设定系统参数需要的时间为18.760毫秒、生成无证书环境的密钥需要的时间为18.554毫秒、生成基于身份环境的密钥需要的时间为12.422毫秒、签密需要的时间为43.373毫秒、解签密需要的时间为62.030毫秒。对于第二种实施方式来说，设定系统参数需要的时间为24.803毫秒、生成无证书环境的密钥需要的时间为19.039毫秒、生成基于身份环境的密钥需要的时间为 12.979毫秒、签密需要的时间为51.687毫秒、解签密需要的时间为62.766毫秒。可见本发明具有处理效率高效和实用的特点。

[0089] 以上所述，仅为本发明的具体实施方式，本说明书中所公开的任一特征，除非特别叙述，均可被其他等效或具有类似目的的替代特征加以替换；所公开的所有特征、或所有方法或过程中的步骤，除了互相排斥的特征和/或步骤以外，均可以任何方式组合。

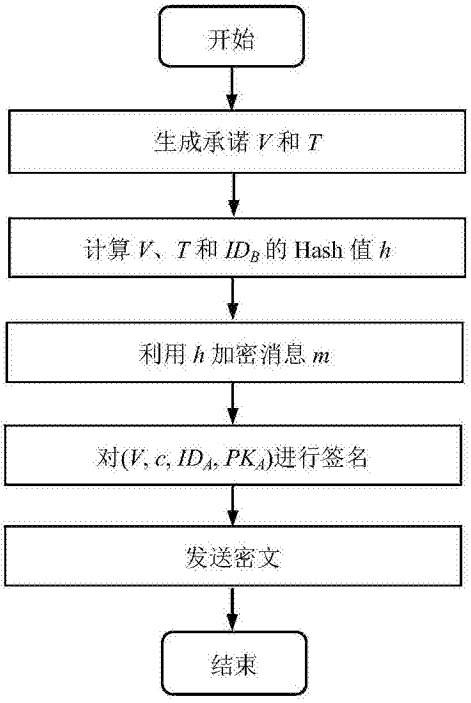

开始
生成承诺 V 和 T
计算 V、T 和 ID_B 的 Hash 值 h
利用 h 加密消息 m
对(V, c, ID_A, PK_A)进行签名
发送密文
结束

图1

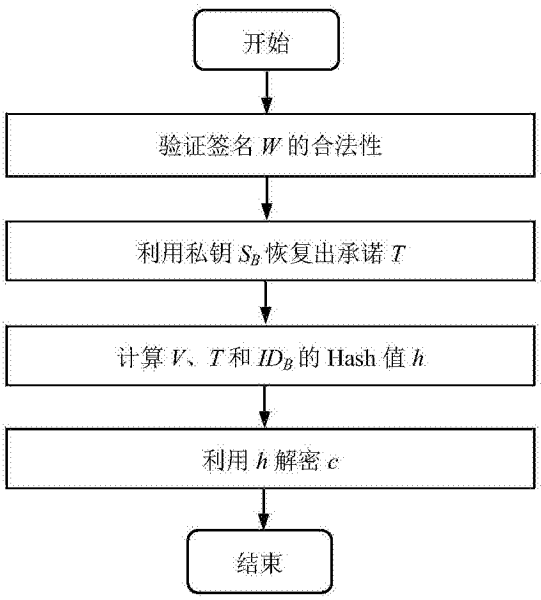

开始
验证签名 W 的合法性
利用私钥 S_B 恢复出承诺 T
计算 V、T 和 ID_B 的 Hash 值 h
利用 h 解密 c
结束

图2

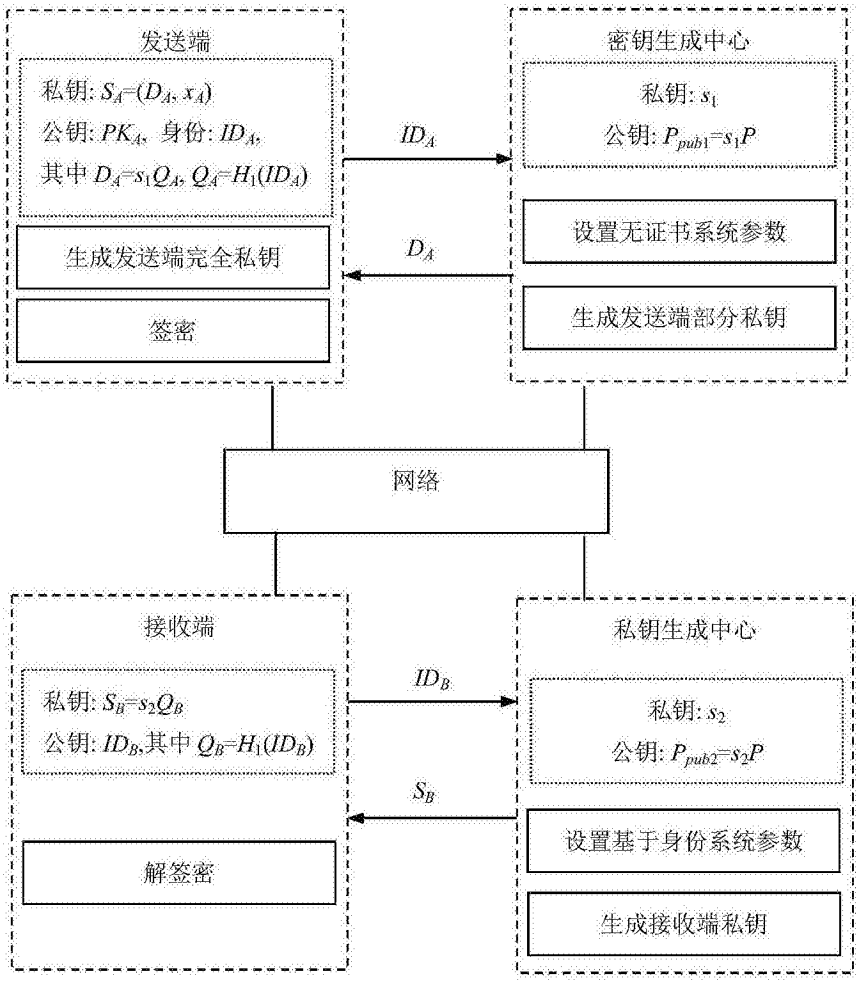

发送端
私钥: $S_A=(D_A, x_A)$
公钥: PK_A, 身份: ID_A,
其中 $D_A=s_1Q_A$, $Q_A=H_1(ID_A)$
生成发送端完全私钥
签密
密钥生成中心
私钥: s_1
公钥: $P_{pub1}=s_1P$
设置无证书系统参数
生成发送端部分私钥
ID_A
D_A
网络
接收端
私钥: $S_B=s_2Q_B$
公钥: ID_B,其中 $Q_B=H_1(ID_B)$
解签密
私钥生成中心
私钥: s_2
公钥: $P_{pub2}=s_2P$
设置基于身份系统参数
生成接收端私钥
ID_B
S_B

图3

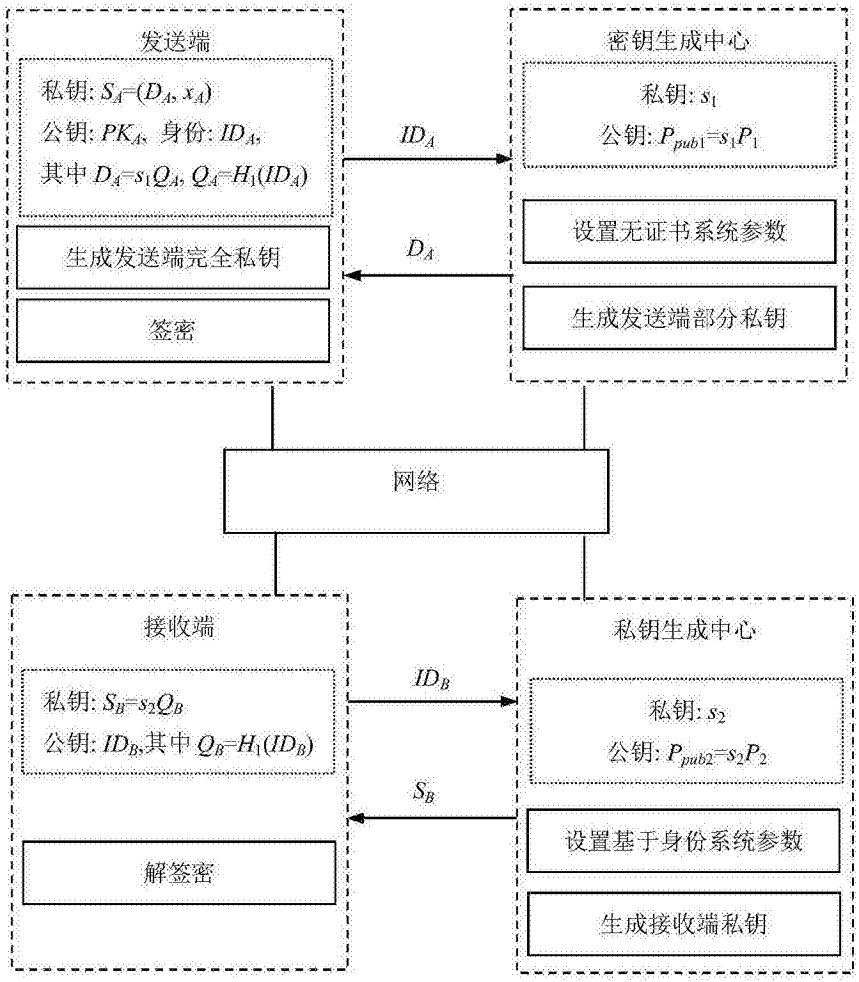

发送端
私钥: $S_A=(D_A, x_A)$
公钥: PK_A, 身份: ID_A,
其中 $D_A=s_1Q_A$, $Q_A=H_1(ID_A)$
生成发送端完全私钥
签密
密钥生成中心
私钥: s_1
公钥: $P_{pub1}=s_1P_1$
设置无证书系统参数
生成发送端部分私钥
ID_A
D_A
网络
接收端
私钥: $S_B=s_2Q_B$
公钥: ID_B,其中 $Q_B=H_1(ID_B)$
解签密
私钥生成中心
私钥: s_2
公钥: $P_{pub2}=s_2P_2$
设置基于身份系统参数
生成接收端私钥
ID_B
S_B

图4