(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 106961336 A
(43)申请公布日 2017.07.18

(21)申请号 201710253749.0

(22)申请日 2017.04.18

(71)申请人 北京百旺信安科技有限公司
地址 100094 北京市海淀区上地三街9号C座8层C905
申请人 熊荣华

(72)发明人 熊荣华
其他发明人请求不公开姓名

(51)Int.Cl.
H04L 9/32(2006.01)
H04L 9/08(2006.01)

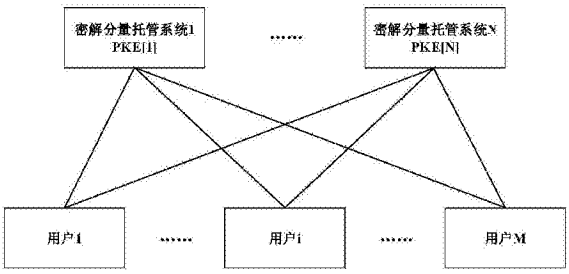
权利要求书3页 说明书7页 附图5页

(54)发明名称

一种基于SM2算法的密钥分量托管方法和系统

(57)摘要

一种基于SM2公钥密码算法的密钥分量托管方法和系统，属于信息安全领域。由N(N≥1)个不同的密钥分量托管系统构成一个密钥托管网络，采用密钥分割存储和多方联合计算的方式，将用户的SM2私钥分割为N+1个分量，由用户和N个不同的密钥分量托管系统分散保存，在需要使用私钥时，由用户和N个密钥分量托管系统联合完成相关计算。在用户密钥生成和使用阶段，密钥分量托管系统通过标识映射算法产生用户私钥分量，但不实际保存，可以大大减小系统建设规模，节约系统投资成本，提高系统运行效率和服务水平。



CN 106961336 A

1.一种基于SM2公钥密码算法的密钥分量托管方法,其特征在于:应用于由N(N≥1)个不同的密钥分量托管系统所构成的一个密钥托管网络;所述方法包括:

采用密钥分割存储和多方联合计算的方式,将用户端的SM2私钥分割为N+1个分量;

由用户端和N个不同的密钥分量托管系统分散保存;

在需要使用私钥对一个消息作数字签名时,由用户端和N个密钥分量托管系统联合完成签名计算,其中,计算形成的签名可使用用户端的公钥进行验证;

在需要对用户端公钥加密后的密文进行解密时,由用户端和N个密钥分量托管系统联合完成对所述密文的解密计算,实现对所述密文的完整解密。

2.根据权利要求1所述的密钥分量托管方法,其特征在于:所述方法还包括:

在每个密钥分量托管系统中生成一个主密钥;

通过一种标识映射算法,在用户密钥生成阶段,密钥分量托管系统采用所生成的主密钥对用户提供的识别信息进行分散映射,生成对应的私钥分量;其中,所述识别信息为:用户标识、用户设置的PIN码和用户端设备信息的一种组合或叠加;

在用户端需使用自己的私钥分量时,按所述标识映射算法恢复用户端的私钥分量后,再进行相关密码运算。

3.根据权利要求1所述的密钥分量托管方法,其特征在于:本发明所依托的密码算法为国密SM2公钥密码算法,与SM2相关的椭圆曲线参数按国密SM2算法标准设置,有限域上的椭圆曲线记为E(Fq),其基点记为G,G的阶记为n;本发明涉及到的实体包括用户端和N(N≥1)个密钥分量托管系统,N个密钥分量托管系统分别记为PKE[1],PKE[2],…,PKE[N];每个用户端针对N个密钥分量托管系统,分别设置N个识别信息UID[1],UID[2],…,UID[N],而用户端数量不限;在系统运行初期,密钥分量托管系统PKE[i]随机生成1个系统主密钥MK[i]和1对SM2非对称密钥,系统主密钥MK[i]也可以用非对称密钥的私钥充当;密钥分量托管系统PKE[i]选取一个密钥分量映射生成函数g(x,y),对任意用户端,通过对系统主密钥MK[i]和用户识别信息UID[i]的映射可生成一个私钥分量$d_i=g(MK[i],UID[i])$,且$d_i\in[1,n-1]$。

4.根据权利要求1所述的密钥分量托管方法,其特征在于:通过以下步骤可完成用户签名密钥的生成与分割存储:

第1步:所述用户端以自己的标识、输入的PIN码和硬件设备信息构造N个用户识别信息UID[1],UID[2],…,UID[N],再设置初始公钥参数$Q_{N+1}=G$;

第2步:对于i=N,N-1,…,1,所述用户端依次与PKE[i]交互执行:

(2a)所述用户端以PKE[i]的公钥对UID[i]加密生成一个密文C[i],发送C[i]和Q_{i+1}到PKE[i];

(2b)所述PKE[i]使用私钥对C[i]解密获取所述用户识别信息UID[i],计算$d_i=g(MK[i],UID[i])$,$Q_i=(d_i)^{-1}Q_{i+1}$,回送Q_i到用户端,将d_i作为用户的第i个签名私钥分量,但不需保存;

(2c)所述用户保存Q_i;

第3步:所述用户端随机选取$d_0\in[1,n-1]$,计算$Q_0=(d_0)^{-1}Q_1$,$Q=Q_0-G$,将d_0作为用户端的签名私钥分量保存,将Q作为用户端的实际签名公钥保存,同时保存Q_0;

通过上述步骤生成的用户实际签名私钥$d=(d_0d_1d_2\cdots d_N)^{-1}-1$,实际签名公钥$Q=dG$,实际签名私钥在生成过程中并未出现,且对用户和N个密钥分量托管系统都不可知。

5.根据权利要求1所述的密钥分量托管方法，其特征在于：通过以下步骤可完成用户加密密钥的生成与分割存储：

第1步：所述用户以自己的标识、输入的PIN码和硬件设备信息构造一个用户识别信息，并将所述用户识别信息记为UID'[1]，UID'[2]，…，UID'[N]，再设置初始值$Q'_{N+1}=G$；

第2步：对于i＝N,N-1，…，1，所述用户依次与PKE[i]交互执行：

(2a)所述用户以PKE[i]的公钥对UID'[i]加密生成一个密文C[i]，发送C[i]和Q'_{i+1}到PKE[i]；

(2b)所述PKE[i]使用私钥对C[i]解密获取所述用户UID'[i]，计算$d_i'=g(MK[i],UID'[i])$，$Q'_i=d_i'Q'_{i+1}$，回送Q'_i到用户端，将d'_i作为用户的第i个加密私钥分量，但不需保存；

第3步：所述用户随机选取$d'_0\in[1,n-1]$，计算$Q'_0=d'_0Q_1$，$Q'=Q'_0$，将d'_0作为用户端的加密私钥分量保存，将Q'作为用户的实际加密公钥保存；

通过上述步骤生成的用户实际加密私钥$d'=d'_0d'_1d'_2\cdots d'_N$，实际加密公钥$Q'=d'G$，实际加密私钥在生成过程中并未出现，且对用户和N个密钥分量托管系统都不可知。

6.根据权利要求1所述的密钥分量托管方法，其特征在于：用户端和N个密钥分量托管系统可联合完成对一个消息的SM2签名，设待签名的消息为M，$e=h(Z||M)$是对消息M的摘要值，其中，Z是与用户公钥和用户标识有关的信息，所述多方联合签名步骤如下：

第1步：所述用户端以自己的标识、输入的PIN码和硬件设备信息等重新构造用户的识别信息UID[1]，UID[2]，…，UID[N]，再随机选取$k_0\in[1,n-1]$，计算$R_0=k_0Q_0$；

第2步：对于i＝1,2，…，N，所述用户依次与PKE[i]交互执行：

(2a)用户端发送R_{i-1}和Q_i到PKE[i]；

(2b)PKE[i]随机选取$k_i\in[1,n-1]$，计算$R_i=R_{i-1}+k_iQ_i$，回送R_i到用户；

第3步：设$R_N=(x_1,y_1)$，所述用户计算$r=(e+x_1)\pmod n$，并记$s_{N+1}=r$；

第4步：对于i＝N,N-1，…，1，所述用户依次与PKE[i]交互执行：

(4a)所述用户端以所述PKE[i]的公钥对UID[i]||R_i加密生成一个密文C[i]，发送C[i]和s_{i+1}到PKE[i]；

(4b)所述PKE[i]对C[i]作私钥解密获取UID[i]和R_i'，验证$R_i'=R_i$是否成立，若成立，再计算签名私钥分量$d_i=g(MK[i],UID[i])$和部分签名$s_i=k_i+s_{i+1}d_i\pmod n$，回送所述部分签名$s_i$到用户端；

(4c)所述用户端在收到所述PKE[i]的部分签名s_i后，计算$R'=R_{i-1}+s_iQ_i-rG$，检验$R'=R_N$是否成立，若成立，则接受PKE[i]的部分签名s_i；

第5步：所述用户端计算$s=k_0+s_1d_0-r\pmod n$，生成最终签名(r,s)；

按此步骤生成的签名(r,s)可以使用签名公钥Q按SM2签名的验证算法进行验证。

7.根据权利要求1所述的密钥分量托管方法，其特征在于：用户端和N个密钥分量托管系统可联合完成对一个公钥加密的密文进行解密，设待解密的密文为$C_1||C_2||C_3$，其中C_1是一个椭圆曲线点，所述多方联合解密步骤如下：

第1步：所述用户端以自己的标识、输入的PIN码和硬件设备信息重新构造用户的识别信息UID'[1]，UID'[2]，…，UID'[N]，再随机选取$k_0\in[1,n-1]$，且$k_0\neq(d'_0)^{-1}$，计算$D_{N+1}=k_0C_1$；

第2步：对于i＝N,N-1，…，1，所述用户依次与所述PKE[i]交互执行：

(2a)所述用户端从所述PKE[i]获取一个随机数r[i]；

(2b)所述用户端以所述PKE[i]的公钥对UID'[i]||r[i]加密生成一个密文C[i]，发送D_{i+1}和C[i]到PKE[i]；

(2c)所述PKE[i]对C[i]作私钥解密获取UID'[i]和r'[i]，验证r'[i]＝r[i]是否成立，若成立，计算$d'_i=g(MK[i],UID'[i])$和$D_i=d'_iD_{i+1}$，回送D_i到用户端；

第3步：所述用户端计算$D=(k_0)^{-1}d'_0D_1$；

第4步：所述用户端令$D=(x_2,y_2)$，再按SM2解密算法的后续步骤解出明文。

8.一种密钥分量托管系统，其特征在于：所述密钥分量托管系统由服务单元、密钥生成单元、签名运算单元、解密运算单元构成；服务单元接受用户请求，与用户之间进行安全通信，对用户身份进行确认，为用户提供密钥生成、联合签名和联合解密服务；密钥生成单元采用系统主密钥对用户端提供的识别信息进行一种分散映射，在用户端密钥生成阶段和用户端密钥使用阶段，为用户端生成或恢复私钥分量；签名运算单元执行联合签名运算，以用户的私钥分量对所述消息摘要值进行部分签名运算；解密运算单元执行联合解密运算，以用户的私钥分量对所述数据进行部分解密运算。

一种基于SM2算法的密钥分量托管方法和系统

技术领域

[0001] 本发明涉及基于SM2公钥密码算法和密钥分割存储机制的密钥分量托管方法和系统，属于信息安全领域。

背景技术

[0002] 在互联网和云计算环境下，出现了大量与网络相关的应用，如网上银行、网上支付、网上购物和互联网医疗等，需要进行网上用户身份认证、网上操作确认和用户隐私保护，以保证网络应用的安全性。解决这种安全需求的最佳手段是使用公钥密码技术来实现数字签名和公钥加密。在使用公钥密码技术时，保证所使用私钥的安全则是关键所在。在通常情况下，为了保证私钥的存储安全和使用安全，用于签名和解密的私钥都要求保存在密码设备内，相应的密码运算也在密码设备内执行。所采用的密码设备在服务器端通常为密码机，在客户端为带CPU的USBKEY和IC卡等。但在网络环境和手机移动终端等环境下，使用这些密码设备来保存密钥和执行密码运算就很不方便，因而出现了将密钥保存在手机文件中并在手机上执行密码运算的应用需求。这种软环境给密钥的存储安全和使用安全带来很大隐患。为了提高密钥存储和密码运算的安全性，可以采用密钥分割存储和多方联合计算的方式，将用户的私钥分割为几个分量，各个私钥分量分散保存在不同的密钥托管系统中，在需要使用私钥对一个消息作数字签名时，由多个密钥托管系统联合完成签名计算，最后形成的签名可使用用户端的公钥进行验证。同样地，在需要对用户端公钥加密后的密文进行解密时，由多个密钥托管系统联合完成对密文的解密计算，实现对加密消息的完整解密。针对国密SM2公钥密码算法，已经有了相应的密钥分割方法和联合签名算法(参见申请号为201710157604.0的专利)，现在的需求是要建立安全高效的密钥托管系统，为广大用户提供私钥分量的托管服务。当需要密钥分量托管的用户量很大(例如达到亿级或十亿级水平)时，密钥托管系统必须耗费大量的存储资源和密码设备来保存用户的密钥和与用户相关的信息，必将大大降低系统的运行效率和服务水平。

发明内容

[0003] 本发明针对密钥分割与联合计算需要对大量私钥分量进行托管的应用需求，提出一种基于标识的私钥分量托管方法，并由此构造一种相应的密钥分量托管系统(Partial key escrow，简称PKE)，解决大用户量场景下的用户私钥分量托管问题。采用的技术方案是，建立N(N≥1)个密钥分量托管系统，在每个密钥分量托管系统中生成一个主密钥，通过一种标识映射算法，在用户密钥生成阶段，密钥分量托管系统采用系统主密钥对用户提供的识别信息进行一种分散映射，生成对应的私钥分量，但不实际保存这个私钥分量；在用户需使用自己的私钥分量时，按同样的算法恢复用户的私钥分量后，再进行相关密码运算。这里所指的用户识别信息可以是用户标识、用户设置的PIN码和用户端设备信息的一种组合或叠加。采用这种方案的有益效果是密钥分量托管系统不需要数据库软件系统和大量的密钥存储设备，对管理的用户数量没有限制，可以大大提高系统的运行效率和服务水平，并且

安全性没有任何降低。

[0004] 本发明所述基于SM2算法的密钥分量托管方法和系统所依托的密码算法为国密SM2公钥密码算法和SM3杂凑算法，与SM2相关的椭圆曲线参数按国密 SM2算法标准设置。有限域上的椭圆曲线记为E(Fq)，其基点记为G，G的阶记为n，SM3杂凑函数记作h(x)。

[0005] 本发明所述基于SM2算法的密钥分量托管方法和系统涉及到的实体包括用户和N(N≥1)个密钥分量托管系统，N个密钥分量托管系统分别记为PKE[1]， PKE[2]，…，PKE[N]。在系统运行初期，密钥分量托管系统PKE[i]随机生成1 个系统主密钥MK[i]和1对SM2非对称密钥，系统主密钥MK[i]也可以用非对称密钥的私钥充当。每个用户面对N个密钥分量托管系统，针对每个密钥分量托管系统，用户需要提供不同的用户识别信息，一个用户的N个识别信息分别记为UID[1]，UID[2]，…，UID[N]。

[0006] 所述密钥分量托管系统PKE[i]使用一个密钥分量映射生成函数g(x,y)，对于每个用户，通过对系统主密钥MK[i]和用户识别信息UID[i]的映射可生成一个私钥分量di＝g(MK[i],UID[i])，且di∈[1,n-1]。

[0007] 用户密钥分量托管的具体实现过程分为密钥生成和密钥使用两个阶段，而密钥生成又包含签名密钥生成和加密密钥生成，密钥使用又包含联合签名和联合解密。

[0008] 一、签名密钥生成

[0009] 签名密钥生成由所述用户和所述N个密钥分量托管系统协同完成，生成步骤如下：

[0010] 第1步：所述用户以自己的标识、输入的PIN码和硬件设备信息等构造N个用户识别信息UID[1]，UID[2]，…，UID[N]，再设置初始值QN+1＝G。

[0011] 第2步：对于i＝N,N-1,…,1，所述用户依次与PKE[i]交互执行：

[0012] (2a)所述用户以PKE[i]的公钥对UID[i]加密生成一个密文C[i]，发送C[i] 和Qi+1到PKE[i]。

[0013] (2b)所述PKE[i]使用私钥对C[i]解密获取所述用户UID[i]，计算di＝ g(MK[i]，UID[i])，Qi＝(di)-1Qi+1，回送Qi到用户端，将di作为用户的第i个签名私钥分量，但不需保存。

[0014] (2c)所述用户保存Qi。

[0015] 第3步：所述用户随机选取d0∈[1,n-1]，计算Q0＝(d0)-1Q1，Q＝Q0-G，将d0作为用户端的签名私钥分量保存，将Q作为用户的实际签名公钥保存，同时保存Q0。

[0016] 通过上述步骤生成的用户实际签名私钥d＝(d0d1d2…dN)-1-1，实际签名公钥Q＝dG。实际签名私钥在生成过程中并未出现，且对用户和N个密钥分量托管系统都不可知。用户和N个密钥分量托管系统对各自生成的签名私钥分量具有完全自主权，其他用户或任何第三方都不能获取其签名私钥分量的信息。

[0017] 二、加密密钥生成

[0018] 所述加密密钥主要用于对消息作公钥加密和私钥解密，由所述用户和所述 N个密钥分量托管系统协同完成，生成步骤如下：

[0019] 第1步：所述用户以自己的标识、输入的PIN码和硬件设备信息等构造N个用户识别信息UID’[1]，UID’[2]，…，UID’[N]，再设置初始值Q’N+1＝G。

[0020] 第2步：所述用户设置初始值Q’N+1＝G，对于i＝N,N-1,…,1，依次与PKE[i] 交互执行：

[0021] (2a)所述用户以PKE[i]的公钥对UID'[i]加密生成一个密文C[i],发送 C[i]和Q'i+1到PKE[i]。

[0022] (2b)所述PKE[i]使用私钥对C[i]解密获取所述用户UID'[i],计算d'i= g(MK[i],UID'[i]),Q'i=d'iQ'i+1,回送Q'i到用户端,将d'i作为用户的第i个加密私钥分量,但不需保存。

[0023] 第3步:所述用户随机选取d'0∈[1,n-1],计算Q'0=d'0Q1,Q'=Q'0,将d'0作为用户端的加密私钥分量保存,将Q'作为用户的实际加密公钥保存。

[0024] 通过上述步骤生成的用户实际加密私钥d'=d'0d'1d'2…d'N,实际加密公钥Q'=d'G。实际加密私钥在生成过程中并未出现,且对用户和N个密钥分量托管系统都不可知。用户和N个密钥分量托管系统对各自生成的私钥分量具有完全自主权,其他用户不能获取其私钥分量的信息。

[0025] 三、联合签名

[0026] 本发明所述联合签名是指用户需要对消息作数字签名时,由用户和N个密钥分量托管系统联合按序完成对消息的签名,且签名结果为符合SM2标准的普通签名,签名接收方可以使用用户的实际签名公钥进行验证。

[0027] 所述签名方法涉及到用户和N个密钥分量托管系统。所述用户具有签名私钥分量d0,所述密钥分量托管系统PKE[i]具有签名私钥分量di,所述用户的实际签名公钥为Q=((d0d1…dN)-1-1)G。

[0028] 设待签名的消息为M,e=h(Z||M)是对消息M的摘要值,其中Z是与用户公钥和用户标识有关的信息。首先由用户和N个密钥分量托管系统按次序联合生成一对随机密钥对,再按相反的次序联合完成对摘要值e的SM2签名。所述多方联合签名步骤如下。

[0029] 第1步:所述用户以自己的标识、输入的PIN码和硬件设备信息等重新构造 N个用户的识别信息UID[1],UID[2],…,UID[N],再选择随机数k0,计算R0 =k0Q0。

[0030] 第2步:对于i=1,2,…,N,所述用户依次与PKE[i]交互执行:

[0031] (2a)用户发送Ri-1和Qi到PKE[i];

[0032] (2b)PKE[i]随机选取ki∈[1,n-1],计算Ri=Ri-1+kiQi,回送Ri到用户;

[0033] 第3步:设RN=(x1,y1),所述用户计算r=(e+x1)(mod n),并记sN+1=r。

[0034] 第4步:对于i=N,N-1,…,1,所述用户依次与PKE[i]交互执行:

[0035] (4a)所述用户以所述PKE[i]的公钥对UID[i]|Ri加密生成一个密文C[i],发送C[i]和si+1到PKE[i];

[0036] (4b)所述PKE[i]对C[i]作私钥解密获取UID[i]和Ri',验证Ri'=Ri是否成立,若成立,再计算签名私钥分量di=g(MK[i],UID[i])和部分签名si=ki+ si+1di(mod n),回送所述部分签名si到用户端;

[0037] (4c)所述用户在收到所述PKE[i]的部分签名si后,计算R'=Ri-1+siQi -rG,检验R'=RN是否成立,若成立,则接受PKE[i]的部分签名si。

[0038] 第5步:所述用户计算s=k0+s1d0-r(mod n),生成最终签名(r,s)。

[0039] 按此步骤生成的签名(r,s)可以使用公钥Q按SM2签名的验证算法进行验证,具体的证明过程可参见申请号为201710157604.0的专利。

[0040] 四、联合解密

[0041] 当用户需要对使用用户的加密公钥加密的密文解密时，可由用户和N个密钥分量托管系统联合完成对密文的解密。

[0042] 所述解密方法涉及到用户和N个密钥分量托管系统。用户具有加密私钥分量d'0，密钥分量托管系统PKE[i]具有加密私钥分量d'i，用户的实际加密私钥为d'＝d'0d'1…d'N(mod n)，实际加密公钥为Q'＝(d'0d'1…d'N)G。

[0043] 设待解密的密文为C1||C2||C3，其中C1是一个椭圆曲线点。按照SM2解密算法，用户需要首先计算D＝d'C1，然后再进行其它解密步骤。由用户和N个密钥分量托管系统联合完成解密的关键点就是联合计算D＝d'C1。所述多方联合解密方案的步骤如下。

[0044] 第1步：所述用户以自己的标识、输入的PIN码和硬件设备信息等重新构造用户的识别信息UID'[1]，UID'[2]，…，UID'[N]，再随机选取k0∈[1,n-1]，且k0≠(d'0)-1，计算DN+1＝k0C1。

[0045] 第2步：对于i＝N,N-1,…,1，所述用户依次与所述PKE[i]交互执行：

[0046] (2a)所述用户从所述PKE[i]获取一个随机数r[i]。

[0047] (2b)所述用户以所述PKE[i]的公钥对UID'[i]||r[i]加密生成一个密文C[i]，发送Di+1和C[i]到PKE[i]。

[0048] (2c)所述PKE[i]对C[i]作私钥解密获取UID'[i]和r'[i]，验证r'[i]＝r[i] 是否成立，若成立，计算d'i＝g(MK[i],UID'[i])(mod n)和Di＝d'iDi+1，回送Di到用户端。

[0049] 第3步：所述用户计算D＝(k0)-1d'0D1。

[0050] 第4步：由于(k0)-1d'0D1＝(k0)-1(d'0d'1…d'N)k0C1＝d'C1，所以通过联合计算得出的D正是私钥解密所需要的计算结果，再按SM2解密算法的后续步骤即可解出明文。

附图说明

[0051] 为了更清楚地说明本发明实施例或现有技术中的技术方案，下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本发明的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的附图。

[0052] 图1为本发明所述密钥分量托管系统与用户端关系结构图。

[0053] 图2为本发明所述密钥分量托管系统内部结构图。

[0054] 图3为本发明所述签名密钥生成流程图。

[0055] 图4为本发明所述加密密钥生成流程图。

[0056] 图5为本发明所述联合签名流程图。

[0057] 图6为本发明所述联合解密流程图。

具体实施方式

[0058] 下面将结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本发明一部分实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本发明保护的范围。

[0059] 本发明提出了一种基于SM2算法的密钥分量托管方法和系统，下面根据附图详细

说明本发明的具体实施方式。

[0060] 本发明所述基于SM2算法的密钥分量托管方法和系统所依托的密码算法为国密SM2公钥密码算法和SM3杂凑算法，与SM2相关的椭圆曲线参数按国密SM2算法标准设置。有限域上的椭圆曲线记为E(Fq)，其基点记为G，G 的阶记为n，SM3杂凑函数记作h(x)。

[0061] 图1所示为基于SM2算法的密钥分量托管系统的整体结构图。整个系统由N(N≥1)个密钥分量托管系统和众多用户端组成，用户端个数不受限制。每个用户端面对N个密钥托管系统，用户端私钥在生成时即被分割成N+1个分量，分别由用户端和N个密钥分量托管系统管理。密钥分量托管系统所管理的用户端私钥分量并不实际保存，而是通过一种映射算法，由密钥分量托管系统的系统主密钥和用户端的识别信息计算得出，在密钥生成阶段确定，在密钥使用阶段重新计算后恢复，使用完成后自动销毁。这种密钥分量托管方式的有益效果是不需要实际存储用户端私钥分量，节省大量的存储资源和密码设备，对用户端数量没有限制，大大提高系统的运行效率和服务水平。

[0062] 图2所示是密钥分量托管系统的内部结构。所述密钥分量托管系统由服务单元、密钥生成单元、签名运算单元、解密运算单元构成。服务单元接受用户端请求，与用户端之间进行安全通信，对用户端身份进行确认，为用户端提供密钥生成、联合签名和联合解密服务。密钥生成单元采用系统主密钥对用户端提供的识别信息进行一种分散映射，在用户端密钥生成阶段和用户端密钥使用阶段，为用户端生成或恢复私钥分量。签名运算单元执行联合签名运算，以用户端的私钥分量对所述消息摘要值进行部分签名运算。解密运算单元执行联合解密运算，以用户端的私钥分量对所述数据进行部分解密运算。

[0063] 为帮助理解本发明的实质内容，图3到图6给出了具体的密钥生成和密钥使用的实施步骤，并给出一个具体的映射函数g(MK,UID)＝h(MK||UID)(mod n)，其中h(x)为SM3杂凑函数。

[0064] 图3所示为本发明所述签名密钥生成流程，包括以下几个步骤：

[0065] 步骤(1)：所述用户端设置用户端识别信息UID[i]，再设置初始公钥参数 QN+1＝G。

[0066] 步骤(2)：所述用户端以PKE[N]的公钥对UID[N]加密，发送加密的UID[N] 和QN+1到PKE[N]。PKE[N]使用私钥解密获取所述用户端UID[N]，计算dN＝ h(MK[N]||UID[N])(mod n)，QN＝(dN)-1QN+1，回送QN到用户端，将dN作为用户端的第N个签名私钥分量，但不需要保存。

[0067] 步骤(3)：对于i＝N-1，…，2，所述用户端以PKE[i]的公钥对UID[i]加密，发送加密的UID[i]和Qi+1到PKE[i]。PKE[i]使用私钥解密获取所述用户端 UID[i]，计算di＝h(MK[i]||UID[i])(mod n)，Qi＝(di)-1Qi+1，回送Qi到用户端，将di作为用户端的第i个签名私钥分量，但不需要保存。

[0068] 步骤(4)：所述用户端以PKE[1]的公钥对UID[1]加密，发送加密的UID[1] 和Q2到PKE[1]。PKE[1]使用私钥解密获取所述用户端UID[1]，计算d1＝h(MK[1]||UID[1])(mod n)，Q1＝(d1)-1Q2，回送Q1到用户端，将d1作为用户端的第1个签名私钥分量，但不需要保存。

[0069] 步骤(5)：所述用户端随机选取d0∈[1,n-1]，计算Q0＝(d0)-1Q1，Q＝Q0-G，将d0作为用户端的签名私钥分量保存，将Q作为用户端的实际签名公钥保存，同时保存{Q0,Q1,…,

QN}。

[0070] 图4所示为本发明所述加密密钥生成流程，包括以下几个步骤：

[0071] 步骤(6)：所述用户端以自己的标识、输入的PIN码和硬件设备信息等构造一组用户端识别信息，并将所述用户端识别信息记为UID'[i]，再设置初始值Q'N+1＝G。

[0072] 步骤(7)：所述用户端以PKE[N]的公钥对UID'[N]加密，发送加密的UID'[N]和Q'N+1到PKE[N]。PKE[N]使用私钥解密获取所述用户端UID'[N]，计算d'N＝h(MK[N]||UID'[N])(mod n)，Q'N＝d'NQN+1，回送Q'N到用户端，将d'N作为用户端的第N个加密私钥分量，但不需要保存。

[0073] 步骤(8)：对于i＝N-1，…，2，所述用户端以PKE[i]的公钥对UID'[i]加密，发送加密的UID'[i]和Q'i+1到PKE[i]。PKE[i]使用私钥解密获取所述用户端UID'[i]，计算d'i＝h(MK[i]||UID'[i])(mod n)，Q'i＝d'iQi+1，回送Q'i到用户端，将d'i作为用户端的第i个加密私钥分量，但不需要保存。

[0074] 步骤(9)：所述用户端以PKE[1]的公钥对UID'[1]加密，发送加密的UID'[1]和Q'2到PKE[1]。PKE[1]使用私钥解密获取所述用户端UID'[1]，计算d'1＝h(MK[1]||UID'[1])(mod n)，Q'1＝d'1Q2，回送Q'1到用户端，将d'1作为用户的第1个加密私钥分量，但不需要保存。

[0075] 步骤(10)：所述用户端随机选取d'0∈[1，n-1]，计算Q'0＝d'0Q1，Q'＝Q'0，将d'0作为用户端的加密私钥分量保存，将Q'作为用户端的实际加密公钥保存。

[0076] 图5所示为本发明所述联合签名流程，包括以下几个步骤：

[0077] 步骤(11)：所述用户端以自己的标识、输入的PIN码和硬件设备信息等重新构造用户端的识别信息UID[i]，再随机选取k0∈[1，n-1]，计算R0＝k0Q0。设待签名的消息为M，所述用户端计算摘要值e＝h(Z||M)，其中Z为所述用户端的标识信息和公钥信息。

[0078] 步骤(12)：所述用户端发送R0和Q1到PKE[1]，PKE[1]随机选取k1∈[1，n-1]，计算R1＝R0+k1Q1，回送R1到用户端。

[0079] 步骤(13)：对于i＝2，…，N-1，所述用户端发送Ri-1和Qi到PKE[i]，PKE[i]随机选取ki∈[1，n-1]，计算Ri＝Ri-1+kiQi，回送Ri到用户端。

[0080] 步骤(14)：所述用户端发送RN-1和QN到PKE[N]，PKE[N]随机选取kN∈[1，n-1]，计算RN＝RN-1+kNQN，回送RN到用户端。

[0081] 步骤(15)：设RN＝(x1，y1)，所述用户端计算r＝(e+x1)(mod n)。

[0082] 步骤(16)：所述用户端以所述PKE[N]的公钥对UID[N]||RN加密，连同r发送到PKE[N]。所述PKE[N]用私钥解密获取UID[N]和Ri'，验证R'N＝RN是否成立，若成立，再计算签名私钥分量dN＝h(MK[N]||UID[N])(mod n)和部分签名sN＝kN+rdi(mod n)，回送所述部分签名sN到用户端。

[0083] 步骤(17)：所述用户端在收到所述PKE[N]的部分签名sN后，计算R'＝RN-1+sNQN-rG，检验R'＝RN是否成立，若成立，则接受所述PKE[N]的部分签名sN。

[0084] 步骤(18)：对于i＝N-1，…，2，所述用户端以所述PKE[i]的公钥对UID[i]||Ri加密，连同si+1发送到PKE[i]。所述PKE[i]用私钥解密获取UID[i]和Ri'，验证Ri'＝Ri是否成立，若成立，再计算签名私钥分量di＝h(MK[i]||UID[i])(mod n)和部分签名si＝ki+si+1di(mod n)，回送所述部分签名si到用户端。所述用户端在收到所述PKE[i]的部分签名si

后，计算R’＝Ri-1+siQi-rG，检验 R’＝RN是否成立，若成立，则接受PKE[i]的部分签名si。

[0085] 步骤(19)：所述用户端以所述PKE[1]的公钥对UID[1]||R1加密，连同s2 发送到PKE[1]。所述PKE[1]用私钥解密获取UID[1]和R1’，验证R1’＝R1是否成立，若成立，再计算签名私钥分量d1＝h(MK[1]||UID[1])(mod n)和部分签名s1＝k1+s2d1(mod n)，回送所述部分签名s1到用户端。

[0086] 步骤(20)：所述用户端在收到所述PKE[1]的部分签名s1后，计算R’＝R0 +s1Qi-rG，检验R’＝RN是否成立，若成立，则接受PKE[1]的部分签名s1。所述用户端再计算s＝k0+s1d0-r(mod n)，生成最终签名(r,s)。

[0087] 图6所示为本发明所述联合解密流程，包括以下几个步骤：

[0088] 步骤(21)：所述用户端以自己的标识、输入的PIN码和硬件设备信息等重新构造用户端的识别信息UID’[i]，再随机选取k0∈[1,n-1]，且k0≠(d’0)-1，计算DN+1＝k0C1。

[0089] 步骤(22)：所述用户端以所述PKE[N]的公钥对UID’[N]加密，连同DN+1 发送到PKE[N]。所述PKE[N]用私钥解密获取UID’[N]，计算d’N＝ h(MK[N]||UID’[N])(mod n)和DN＝d’N DN+1，回送DN到用户端。

[0090] 步骤(23)：对于i＝N-1，…，2，所述用户端以所述PKE[i]的公钥对UID’[i] 加密，连同Di+1发送到PKE[i]。所述PKE[i]用私钥解密获取UID’[i]，计算 d’i＝h(MK[i]||UID’[i])(mod n)和Di＝d’iDi+1，回送Di到用户端。

[0091] 步骤(24)：所述用户端以所述PKE[1]的公钥对UID’[1]加密，连同D2发送到PKE[1]。所述PKE[1]用私钥解密获取UID’[1]，计算d’1＝h(MK[1]||UID’ [1])(mod n)和D1＝d’1D2，回送D1到用户端。

[0092] 步骤(25)：所述用户端计算D＝(k0)-1d’0D1。

[0093] 步骤(26)：所述用户端按SM2解密算法的后续步骤解出明文。

[0094] 上述实施例仅从原理上描述了本发明的内容，应理解，此处给出的实施例和映射函数仅用于说明本发明的基本思想，并不能限制本发明所具有的一般性。任何对本发明实质内容所作的数学上的变形和修饰都包含在本发明专利的保护范围。

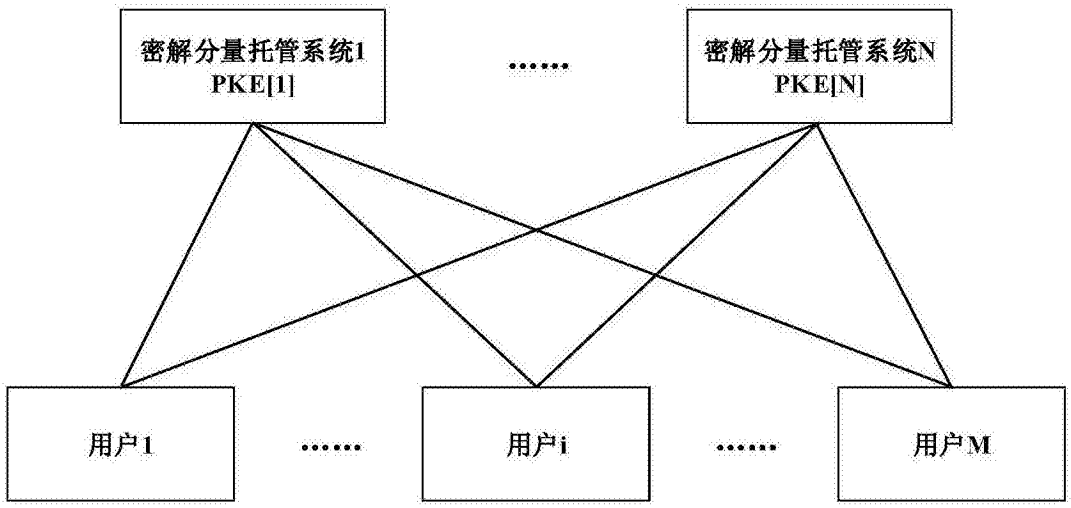

密解分量托管系统1
PKE[1]
……
密解分量托管系统N
PKE[N]
用户1
……
用户i
……
用户M

图1

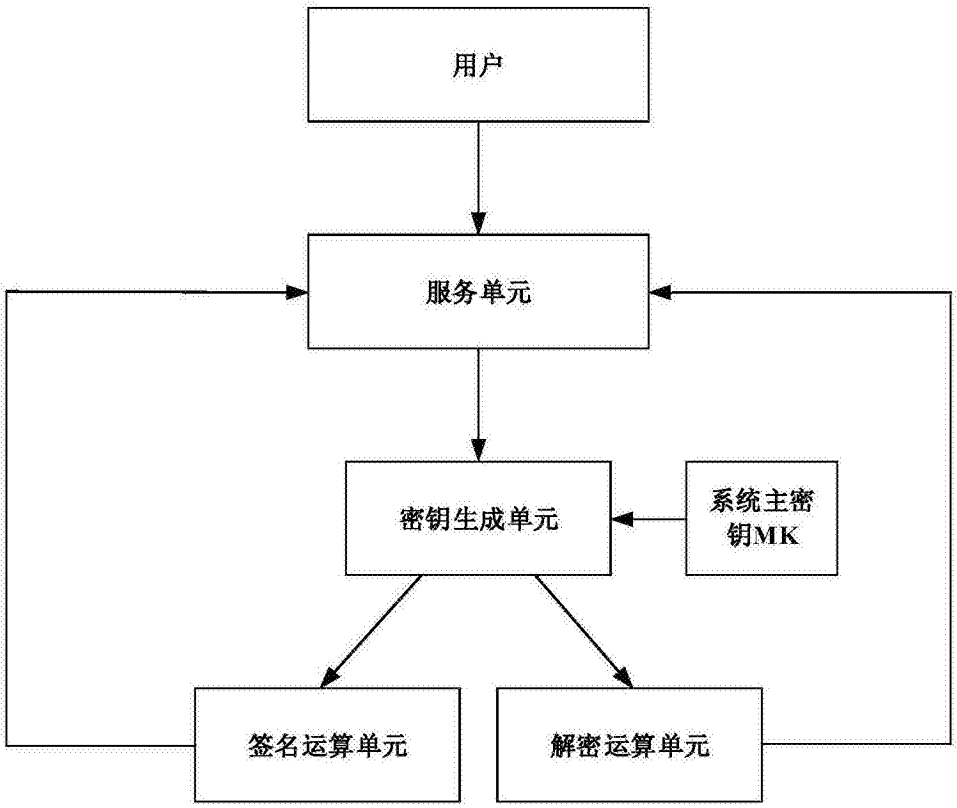

用户
服务单元
密钥生成单元
系统主密钥MK
签名运算单元
解密运算单元

图2

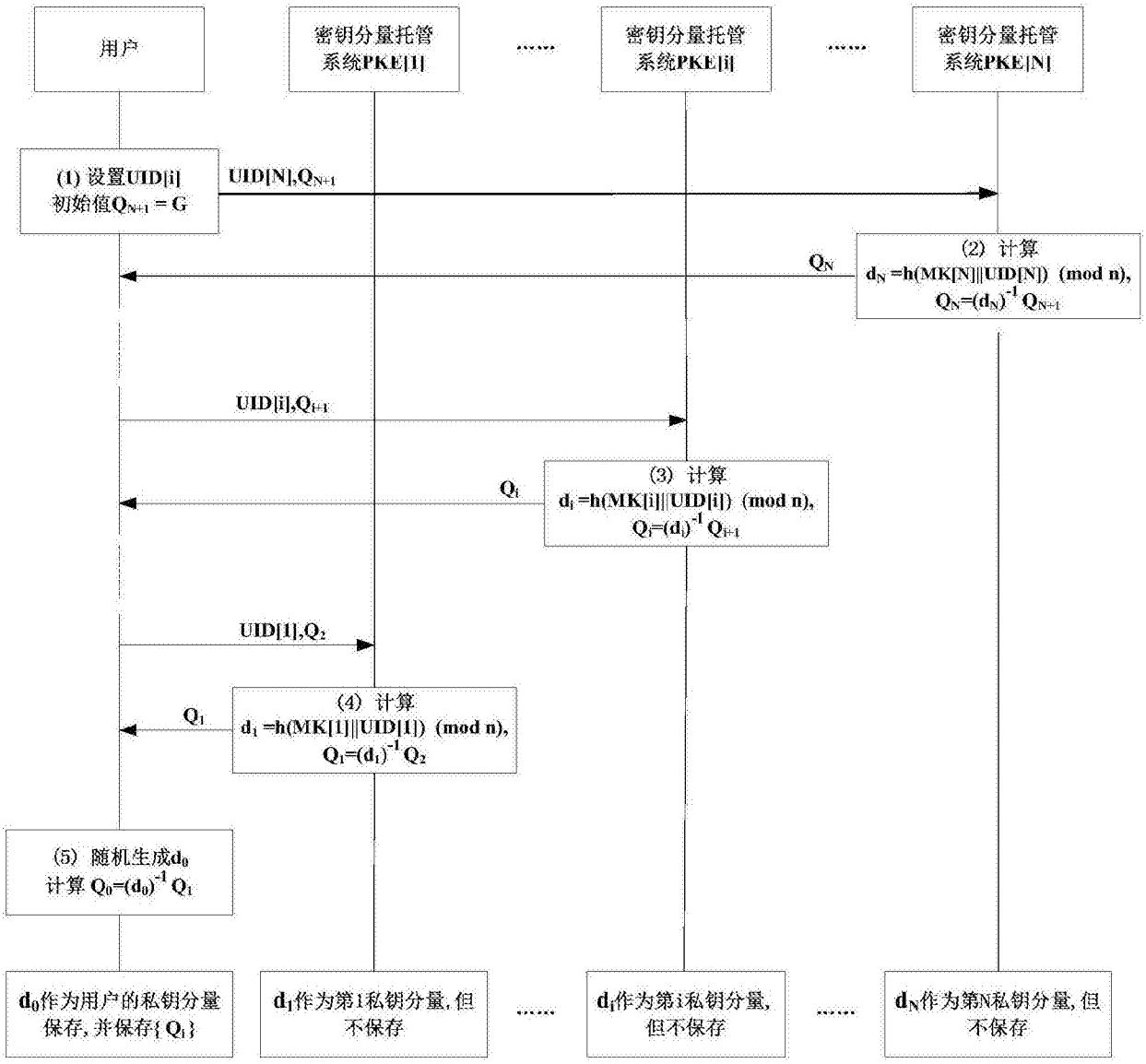

用户
密钥分量托管系统PKE[1]
……
密钥分量托管系统PKE[i]
……
密钥分量托管系统PKE[N]
(1) 设置UID[i]初始值$Q_{N+1}=G$
UID[N],Q_{N+1}
Q_N
(2) 计算 $d_N=h(MK[N]||UID[N])\ (mod\ n)$, $Q_N=(d_N)^{-1}Q_{N+1}$
UID[i],Q_{i+1}
Q_i
(3) 计算 $d_i=h(MK[i]||UID[i])\ (mod\ n)$, $Q_i=(d_i)^{-1}Q_{i+1}$
UID[1],Q_2
Q_1
(4) 计算 $d_1=h(MK[1]||UID[1])\ (mod\ n)$, $Q_1=(d_1)^{-1}Q_2$
(5) 随机生成d_0 计算 $Q_0=(d_0)^{-1}Q_1$
d_0作为用户的私钥分量保存,并保存{Q_1}
d_1作为第1私钥分量,但不保存
……
d_i作为第i私钥分量,但不保存
……
d_N作为第N私钥分量,但不保存

图3

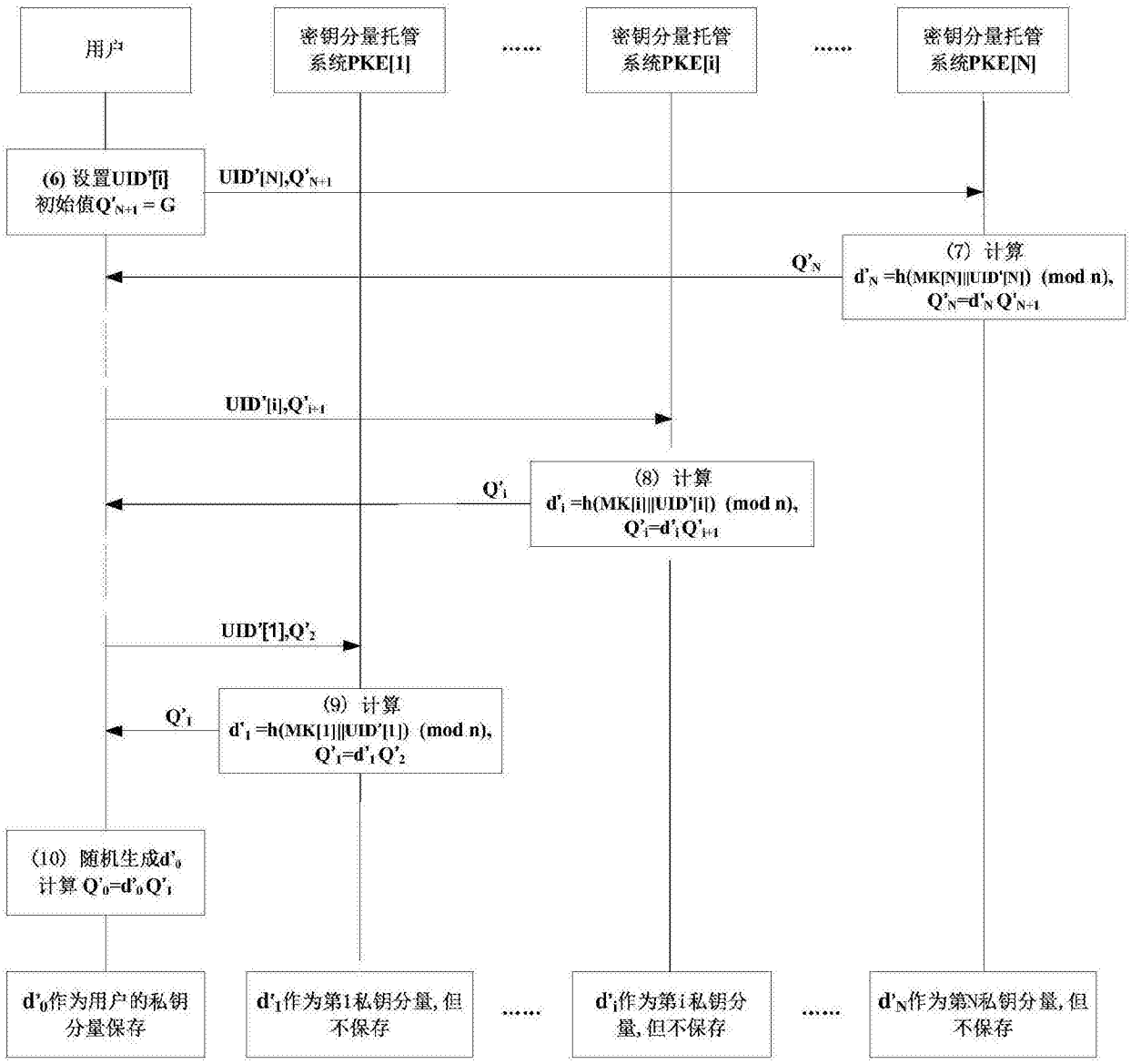

用户
密钥分量托管系统PKE[1]
……
密钥分量托管系统PKE[i]
……
密钥分量托管系统PKE[N]
(6) 设置UID'[i]初始值Q'N+1 = G
UID'[N],Q'N+1
Q'N
(7) 计算
d'N =h(MK[N]||UID'[N]) (mod n),
Q'N=d'N Q'N+1
UID'[i],Q'i+1
Q'i
(8) 计算
d'i =h(MK[i]||UID'[i]) (mod n),
Q'i=d'i Q'i+1
UID'[1],Q'2
Q'1
(9) 计算
d'1 =h(MK[1]||UID'[1]) (mod n),
Q'1=d'1 Q'2
(10) 随机生成d'0
计算 Q'0=d'0 Q'1
d'0作为用户的私钥分量保存
d'1作为第1私钥分量,但不保存
……
d'i作为第i私钥分量,但不保存
……
d'N作为第N私钥分量,但不保存

图4

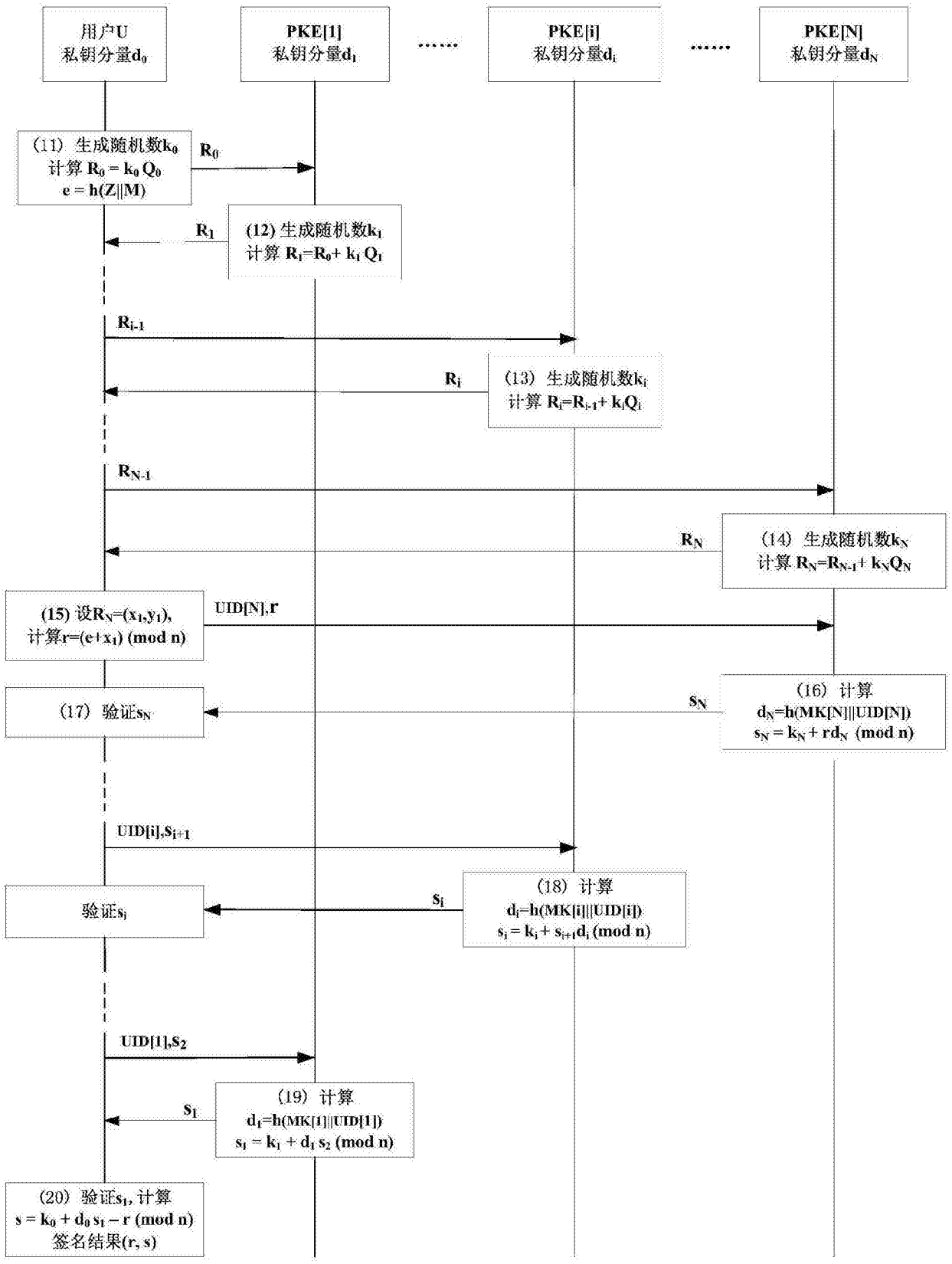

用户U
私钥分量d_0
PKE[1]
私钥分量d_1
……
PKE[i]
私钥分量d_i
……
PKE[N]
私钥分量d_N
(11) 生成随机数k_0
计算 $R_0 = k_0 Q_0$
$e = h(Z||M)$
R_0
R_1
(12) 生成随机数k_1
计算 $R_1=R_0+ k_1 Q_1$
R_{i-1}
R_i
(13) 生成随机数k_i
计算 $R_i=R_{i-1}+ k_iQ_i$
R_{N-1}
R_N
(14) 生成随机数k_N
计算 $R_N=R_{N-1}+ k_NQ_N$
(15) 设$R_N=(x_1,y_1)$,
计算$r=(e+x_1)$ (mod n)
UID[N],r
(17) 验证s_N
s_N
(16) 计算
$d_N=h(MK[N]||UID[N])$
$s_N = k_N + rd_N$ (mod n)
UID[i],s_{i+1}
验证s_i
s_i
(18) 计算
$d_i=h(MK[i]||UID[i])$
$s_i = k_i + s_{i+1}d_i$ (mod n)
UID[1],s_2
s_1
(19) 计算
$d_1=h(MK[1]||UID[1])$
$s_1 = k_1 + d_1 s_2$ (mod n)
(20) 验证s_1, 计算
$s = k_0 + d_0 s_1 - r$ (mod n)
签名结果(r, s)

图5

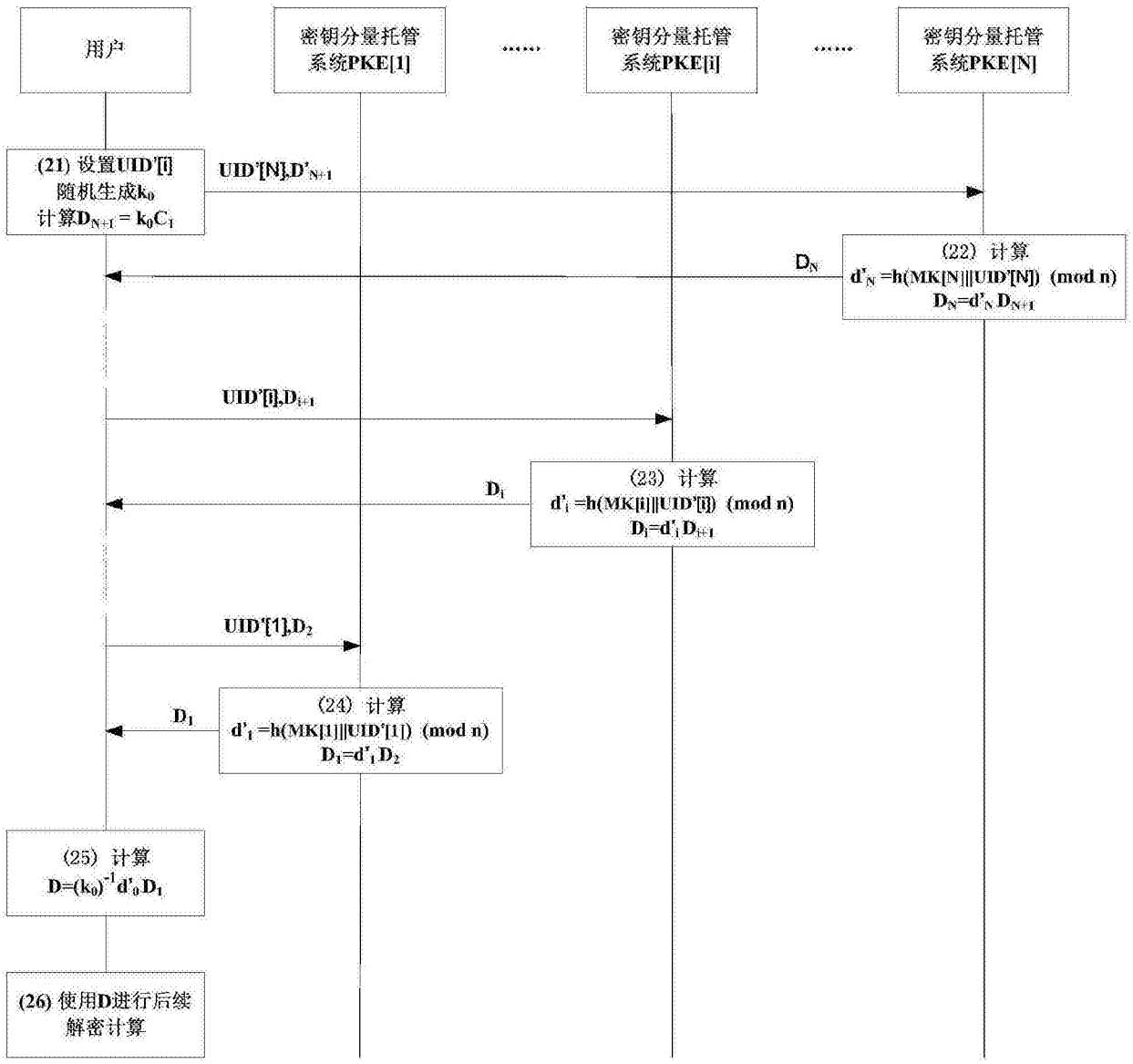

用户
密钥分量托管系统PKE[1]
……
密钥分量托管系统PKE[i]
……
密钥分量托管系统PKE[N]
(21) 设置UID'[i]
随机生成k0
计算DN+1 = k0C1
UID'[N],D'N+1
(22) 计算
d'N =h(MK[N]||UID'[N]) (mod n)
DN=d'N DN+1
DN
UID'[i],Di+1
(23) 计算
d'i =h(MK[i]||UID'[i]) (mod n)
Di=d'i Di+1
Di
UID'[1],D2
(24) 计算
d'1 =h(MK[1]||UID'[1]) (mod n)
D1=d'1 D2
D1
(25) 计算
D=(k0)-1d'0 D1
(26) 使用D进行后续解密计算

图6