(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 108809658 A
(43)申请公布日 2018.11.13

(21)申请号 201810805299.6

(22)申请日 2018.07.20

(71)申请人 武汉大学
地址 430072 湖北省武汉市武昌区八一路299号

(72)发明人 何德彪 张佳妮 陈泌文 张宇波

(74)专利代理机构 湖北武汉永嘉专利代理有限公司 42102
代理人 唐万荣 李丹

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

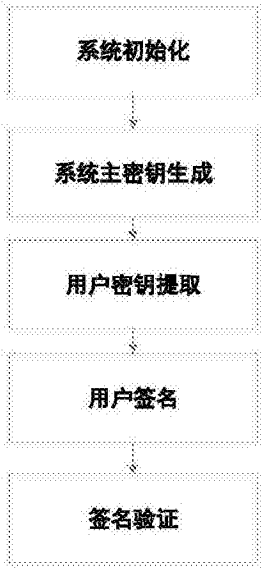
权利要求书4页 说明书9页 附图3页

(54)发明名称

一种基于SM2的身份基的数字签名方法与系统

(57)摘要

本发明公开了一种基于SM2的身份基的数字签名方法与系统，该方法首先产生系统参数、系统主公私钥，用户将身份标识发送给KGC注册以获取签名私钥，然后用户使用签名私钥产生消息M的数字签名并进行验证。由于本发明的签名过程中，在SM2签名算法整体架构不改变的基础上使用新型的用户私钥的产生方式，基于用户的身份来生成用户的签名私钥，不再使用公钥证书，无需维护和管理公钥证书以及耗时计算，从而保证了本发明签名方法具有低复杂度、高安全性、易验证等特点。



CN 108809658 A

1.一种基于SM2的身份基的数字签名方法，其特征在于，包括以下步骤：

步骤1)产生整个签名过程所需参数，参数包括：椭圆曲线相关参数：q、F_q、a、b、n、G，和安全哈希函数：$H_V(\cdot)$、$H(\cdot)$、$H_{256}(\cdot)$；

其中，q为大素数，F_q为包含q个元素的有限域，a、b为F_q中的元素，用于定义F_q上的一条椭圆曲线E；G为椭圆曲线的一个基点，其阶为素数；n为基点G的阶；

步骤2)由密钥生成中心KGC生成系统主公私钥(P_{pub},P_{pri})，其中，P_{pub}为系统主公钥，P_{pri}为系统主私钥；

步骤3)用户密钥提取：产生用户User的签名私钥，具体过程如下：

步骤3.1)用户User将身份标识ID发送给KGC，请求签名私钥；

步骤3.2)KGC收到私钥请求后，首先利用已有的身份认证方法确认ID与User身份一致，随后，KGC在集合{1,2,…,n-1}中随机选取整数l并通过以下公式计算签名私钥的第一部分L；

$L=[l]G=(x_1,y_1)$

其中，(x_1,y_1)表示L的横纵坐标；

步骤3.3)KGC根据L，通过以下公式计算私钥的第二部分α；

$h=H(ID||L)$

$\alpha=l+xh \bmod n$

其中，$H(\cdot)$表示安全哈希函数，符号||表示连接，modn表示模n运算；

步骤3.4)KGC将私钥(L,α)通过安全信道发送给用户User；

步骤3.5)用户User接收并秘密保存KGC发送的私钥(L,α)；

步骤4)用户签名：该步骤主要用于用户User产生消息M的数字签名(L,r,s)；

步骤5)签名验证：验证通过接收的消息M'的签名(L',r',s')合法性，具体验证过程如下：

步骤5.1)检查r'是否属于集合{1,2,…,n-1}中，如果不是则验证不通过；否则，检查s'是否属于{1,2,…,n-1}，如果不是则验证不通过，否则进入步骤5.2)；

步骤5.2)根据以下公式计算Z'_A并置$\overline{M'}=Z'_A\|M'$；

$Z'_A=H_{256}(ENTLA'||ID||a||b||x_G||y_G||x_1||y_1)$

步骤5.3)根据以下公式计算e'，并将e'转换为整数；

$e'=H_v(\overline{M'})$

步骤5.4)根据以下公式计算并判断t，如果t=0成立，则消息签名为非法，结束验证过程，否则进入步骤5.5)；

$t=r'+s' \bmod n$

步骤5.5)根据以下公式先后计算h'和椭圆曲线点K'，

$h'=H(ID||L')$

$K'=(x'_k,y'_k)=s'G+t(L'+h'P_{pub})$

步骤5.6)根据以下公式计算并判断R'，如果R'=r'成立，则消息签名合法，否则消息签名非法；

$R'=(e'+x'_k) \bmod n$。

2.根据权利要求1所述的基于SM2的身份基的数字签名方法，其特征在于，所述步骤2)中，由密钥生成中心KGC生成系统主公私钥的具体过程如下：

步骤2.1)KGC从集合{1,2,…,n-1}中随机选取整数x作为主私钥$P_{pri}=x$，其中n表示SM2密码运算所使用的椭圆曲线点群的阶。

步骤2.2)KGC根据所选主私钥x，通过以下公式计算并公布系统主公钥P_{pub}，$P_{pub}=[x]G=(x_G,y_G)$

其中，G为SM2数字签名系统参数中椭圆曲线点群的基点，[x]G表示点乘运算，(x_G,y_G)表示公钥的横纵坐标。

3.根据权利要求1所述的基于SM2的身份基的数字签名方法，其特征在于，所述步骤4)的签名过程具体如下：

步骤4.1)用户User根据以下公式计算Z_A并置$\overline{M}=Z_A\|M$

$Z_A=H_{256}(ENTLA||ID||a||b||x_G||y_G||x_1||y_1)$

其中，H_{256}表示安全哈希函数，a,b为椭圆曲线参数，ENTLA表示当前用户ID长度；

步骤4.2)用户User根据以下公式计算e并将e转换为整数，

$e=H_v(\overline{M})$

其中，$H_v(\bullet)$表示安全哈希函数；

步骤4.3)用户User在集合{1,2,…,n-1}中随机选择整数k，根据以下公式计算椭圆曲线点K，

$K=[k]G=(x_k,y_k)$

并将x_k转换为整数类型；

步骤4.4)用户User根据以下公式计算部分签名r并判断r，如果r=0或r+k=n成立，则返回步骤4.3，否则执行步骤4.5；

$r=(e+x_k)\ mod n$

步骤4.5)用户User根据以下公式计算部分签名s并判断s，如果s=0成立，则返回步骤4.3，否则执行步骤4.6；

$s=(1+\alpha)^{-1}(k-r\alpha)\ mod n$

其中，$(1+\alpha)^{-1}$为$(1+\alpha)$的模n乘法逆；

步骤4.6)用户User输出消息M签名为(L,r,s)。

4.一种基于SM2的身份基的数字签名系统，其特征在于，包括：

系统初始化模块，用于产生整个签名系统所需参数，参数包括：椭圆曲线相关参数：q、F_q、a、b、n、G，和安全哈希函数：$H_v(\bullet)$、$H(\bullet)$、$H_{256}(\bullet)$；

其中，q为大素数，F_q为包含q个元素的有限域，a、b为F_q中的元素，用于定义F_q上的一条椭圆曲线E；G为椭圆曲线的一个基点，其阶为素数；n为基点G的阶；

系统密钥生成模块，用于由密钥生成中心KGC生成系统主公私钥(P_{pub},P_{pri})，其中，P_{pub}为系统主公钥，P_{pri}为系统主私钥；

用户密钥提取模块，用于产生用户User的签名私钥，具体过程如下：

1)用户User将身份标识ID发送给KGC，请求签名私钥；

2)KGC收到私钥请求后，首先利用已有的身份认证方法确认ID与User身份一致，随后，

KGC在集合{1,2,…,n-1}中随机选取整数l并通过以下公式计算签名私钥的第一部分L；

$L=[l]G=(x_1,y_1)$

其中，(x_1,y_1)表示L的横纵坐标。

3)KGC根据L，通过以下公式计算签名私钥的第二部分α；

$h=H(ID||L)$

$\alpha=l+xh \bmod n$

其中，H(•)表示安全哈希函数，符号||表示连接，modn表示模n运算；

4)KGC将私钥(L,α)通过安全信道发送给用户User；

5)用户User接收并秘密保存KGC发送的私钥(L,α)；

用户签名模块，用于用户User产生消息M的数字签名(L,r,s)；

签名验证模块，用于验证消息M'的签名(L',r',s')合法性。

5.根据权利要求4所述的基于SM2的身份基的数字签名系统，其特征在于，所述系统密钥生成模块中，由密钥生成中心KGC生成系统主公私钥的具体过程如下：

步骤1)KGC从集合{1,2,…,n-1}中随机选取整数x作为主私钥$P_{pri}=x$，其中n表示SM2密码运算所使用的椭圆曲线点群的阶。

步骤2)KGC根据所选主私钥x，通过以下公式计算并公布系统主公钥P_{pub}，

$P_{pub}=[x]G=(x_G,y_G)$

其中，G为SM2数字签名系统参数中椭圆曲线点群的基点，[x]G表示点乘运算，(x_G,y_G)表示公钥的横纵坐标。

6.根据权利要求4所述的基于SM2的身份基的数字签名系统，其特征在于，所述用户签名模块中的签名过程具体如下：

步骤1)用户User根据以下公式计算Z_A并置$\overline{M}=Z_A\|M$

$Z_A=H_{256}(ENTLA||ID||a||b||x_G||y_G||x_1||y_1)$

其中，H_{256}表示安全哈希函数，a,b为椭圆曲线参数，ENTLA表示当前用户ID长度。

步骤2)用户User根据以下公式计算e并将e转换为整数，

$e=H_v(\overline{M})$

其中，H_v(•)表示安全哈希函数；

步骤3)用户User在集合{1,2,…,n-1}中随机选择整数k，根据以下公式计算椭圆曲线点K，

$K=[k]G=(x_k,y_k)$

并将x_k转换为整数类型；

步骤4)用户User根据以下公式计算部分签名r并判断r，如果r=0或r+k=n成立，则返回步骤3)，否则执行步骤5)；

$r=(e+x_k) \bmod n$

步骤5)用户User根据以下公式计算部分签名s并判断s，如果s=0成立，则返回步骤3)，否则执行步骤6)；

$s=(1+\alpha)^{-1}(k-r\alpha) \bmod n$

其中，$(1+\alpha)^{-1}$为(1+α)的模n乘法逆；

步骤6)用户User输出消息M签名为(L,r,s)。

7.根据权利要求4所述的基于SM2的身份基的数字签名系统,其特征在于,所述签名验证模块中验证消息M'的签名(L',r',s')合法性的具体验证过程如下:

步骤1)检查r'是否属于集合{1,2,…,n-1}中,如果不是则验证不通过;否则,检查s'是否属于{1,2,…,n-1},如果不是则验证不通过,否则进入步骤2);

步骤2)根据以下公式计算Z'_A并置$\overline{M'}=Z'_A\|M'$;

$Z'_A=H_{256}(ENTLA'||ID||a||b||x_G||y_G||x_1||y_1)$

步骤3)根据以下公式计算e',并将e'转换为整数;

$e'=H_v(\overline{M'})$

步骤4)根据以下公式计算并判断t,如果t=0成立,则消息签名为非法,结束验证过程,否则进入步骤5);

t=r'+s'modn

步骤5)根据以下公式先后计算h'和椭圆曲线点K',

h'=H(ID||L')

$K'=(x'_k,y'_k)=s'G+t(L'+h'P_{pub})$

步骤6)根据以下公式计算并判断R',如果R'=r'成立,则消息签名合法,否则消息签名非法;

$R'=(e'+x'_k)$modn。

一种基于SM2的身份基的数字签名方法与系统

技术领域

[0001] 本发明涉及信息安全技术领域，尤其涉及一种基于SM2的身份基的数字签名方法与系统。

背景技术

[0002] 数字签名是伴随着信息网络技术的发展而出现的一种安全保障技术，目的就是通过技术手段实现传统的纸面签字或者盖章的功能，用于鉴定签名人的身份以及对一项电子数据内容的认可。它还能验证出文件的原文在传输过程中有无变动，确保传输电子文件的完整性、真实性和不可抵赖性。数字签名是公钥密码体系中重要的一部分，在很多场合有着重要的作用。

[0003] SM2椭圆曲线公钥密码算法是由国家密码管理局颁布的一种椭圆曲线公钥密码算法(参见《SM2椭圆曲线公钥密码算法》规范，国家密码管理局，2010年12月)，算法确定了包括数据加密、数字签名和密钥交换等算法或协议。其中，SM2签名算法因其高安全、高效率而广泛用于消息传输，可有效保证消息传输过程中消息的可靠性。SM2签名算法包括系统初始化算法(SM2_Setup)、用户密钥对生成算法(SM2_KeyGen)、数字签名算法(SM2_Sign)、签名验证算法(SM2_Verify)。由于SM2签名算法是基于PKI框架设计的密码系统，需要证书授权中心(CA)维护管理用户公钥证书，主要包括证书的颁发、撤销等。证书的管理开销随着用户数量增加而成线性增长，因此，在云计算、大数据等多用户环境下，高昂的证书管理开销将限制SM2公钥密码算法的使用。

[0004] 为解决基于PKI框架下证书管理开销大、维护困难等问题，国家于2015年确立SM9标识密码算法为国家密码行业标准(GM/T 0044-2016)。SM9标识密码算法是一种基于双线性对的身份基密码算法。SM9标识算法包括数字签名、数据加密、密钥交换以及身份认证等。在SM9签名算法中，密钥生成中心(KGC)根据用户注册身份利用密钥提取算法(Extract)产生签名私钥，签名验证算法(SM9_Verify)利用用户身份来验证签名正确性，避免了证书管理问题。然而，由于SM9签名算法在产生签名和验证签名过程中，使用了耗时的双线对运算，对于计算能力有限移动设备，执行双线性对运算是无法承担的。

[0005] 基于SM2和SM9签名算法现有问题，SM2签名算法需要承担证书管理的开销，使其无法满足多用户的网络新环境，而SM9签名算法虽然免去了证书管理，由于耗时的双线对运算严重影响其在移动端应用。

发明内容

[0006] 本发明要解决的技术问题在于针对现有技术中的缺陷，提供一种基于SM2的身份基的数字签名方法与系统，解决现有SM2签名算法中证书管理问题和SM9标识签名算法中需要耗时的双线对运算问题。

[0007] 本发明解决其技术问题所采用的技术方案是：一种基于SM2的身份基的数字签名方法，包括以下步骤：

[0008] 步骤1)产生整个签名过程所需参数,参数包括:椭圆曲线相关参数:q、F_q、a、b、n、G,和安全哈希函数:$H_V(\cdot)$、$H(\cdot)$、$H_{256}(\cdot)$;

[0009] 其中,q为大素数,F_q为包含q个元素的有限域,a、b为F_q中的元素,用于定义F_q上的一条椭圆曲线E;G为椭圆曲线的一个基点,其阶为素数;n为基点G的阶;步骤2)由密钥生成中心KGC生成系统主公私钥(P_{pub},P_{pri}),其中,P_{pub}为系统主公钥,P_{pri}为系统主私钥;

[0010] 步骤2)由密钥生成中心KGC生成系统主公私钥(P_{pub},P_{pri}),其中,P_{pub}为系统主公钥,P_{pri}为系统主私钥;

[0011] 步骤3)用户密钥提取:产生用户User的签名私钥,具体过程如下:

[0012] 步骤3.1)用户User将身份标识ID发送给KGC,请求签名私钥;

[0013] 步骤3.2)KGC收到私钥请求后,首先利用已有的身份认证方法确认ID与User身份一致,随后,KGC在集合{1,2,…,n-1}中随机选取整数l并通过以下公式计算签名私钥的第一部分L;

[0014] $L=[l]G=(x_1,y_1)$

[0015] 其中,(x_1,y_1)表示L的横纵坐标。

[0016] 步骤3.3)KGC根据L,通过以下公式计算私钥的第二部分α;

[0017] $h=H(ID||L)$

[0018] $\alpha=l+xh \bmod n$

[0019] 其中,$H(\cdot)$表示安全哈希函数,符号||表示连接,modn表示模n运算;

[0020] 步骤3.4)KGC将私钥(L,α)通过安全信道发送给用户User;

[0021] 步骤3.5)用户User接收并秘密保存KGC发送的私钥(L,α);

[0022] 步骤4)用户签名(Sign):该步骤主要用于用户User产生消息M的数字签名(L,r,s);

[0023] 步骤5)签名验证(Verify):验证消息M'的签名(L',r',s')合法性,具体验证过程如下:

[0024] 步骤5.1)检查r'是否属于集合{1,2,…,n-1}中,如果不是则验证不通过;否则,检查s'是否属于{1,2,…,n-1},如果不是则验证不通过,否则进入步骤5.2;

[0025] 步骤5.2)根据以下公式计算Z'_A并置$\overline{M'}=Z'_A\|M'$;

[0026] $Z'_A=H_{256}(ENTLA'||ID||a||b||x_G||y_G||x_1||y_1)$

[0027] 步骤5.3)根据以下公式计算e',并将e'转换为整数;

[0028] $e'=H_1(\overline{M'})$

[0029] 步骤5.4)根据以下公式计算并判断t,如果t=0成立,则消息签名为非法,结束验证过程,否则进入步骤5.5;

[0030] $t=r'+s' \bmod n$

[0031] 步骤5.5)根据以下公式先后计算h'和椭圆曲线点K',

[0032] $h'=H(ID||L')$

[0033] $K'=(x'_k,y'_k)=s'G+t(L'+h'P_{pub})$

[0034] 步骤5.6)根据以下公式计算并判断R',如果R'=r'成立,则消息签名合法,否则消息签名非法;

[0035] $R'=(e'+x'_k) \bmod n$。

[0036] 按上述方案，所述步骤2)中，由密钥生成中心KGC生成系统主公私钥的具体过程如下：

[0037] 步骤2.1)KGC从集合{1,2,…,n-1}中随机选取整数x作为主私钥$P_{pri}=x$，其中n表示SM2密码运算所使用的椭圆曲线点群的阶。

[0038] 步骤2.2)KGC根据所选主私钥x，通过以下公式计算并公布系统主公钥P_{pub}，

[0039] $P_{pub}=[x]G=(x_G,y_G)$

[0040] 其中，G为SM2数字签名系统参数中椭圆曲线点群的基点，[x]G表示点乘运算，(x_G,y_G)表示公钥的横纵坐标。

[0041] 按上述方案，所述步骤4)的签名过程具体如下：

[0042] 步骤4.1)用户User根据以下公式计算Z_A并置$\overline{M}=Z_A\|M$

[0043] $Z_A=H_{256}(ENTLA||ID||a||b||x_G||y_G||x_1||y_1)$

[0044] 其中，H_{256}表示安全哈希函数，a,b为椭圆曲线参数，ENTLA表示当前用户ID长度。

[0045] 步骤4.2)用户User根据以下公式计算e并将e转换为整数，

[0046] $e=H_v(\overline{M})$

[0047] 其中，$H_V(\cdot)$表示安全哈希函数；

[0048] 步骤4.3)用户User在集合{1,2,…,n-1}中随机选择整数k，根据以下公式计算椭圆曲线点K，

[0049] $K=[k]G=(x_k,y_k)$

[0050] 并将x_k转换为整数类型；

[0051] 步骤4.4)用户User根据以下公式计算部分签名r并判断r，如果r=0或r+k=n成立，则返回步骤4.3，否则执行步骤4.5；

[0052] $r=(e+x_k) \bmod n$

[0053] 步骤4.5)用户User根据以下公式计算部分签名s并判断s，如果s=0成立，则返回步骤4.3，否则执行步骤4.6；

[0054] $s=(1+\alpha)^{-1}(k-r\alpha) \bmod n$

[0055] 其中，$(1+\alpha)^{-1}$为$(1+\alpha)$的模n乘法逆；

[0056] 步骤4.6)用户User输出消息M签名为(L,r,s)。

[0057] 一种基于SM2的身份基的数字签名系统，包括：

[0058] 系统初始化模块，用于产生整个签名系统所需参数，参数包括：椭圆曲线相关参数：q、F_q、a、b、n、G，和安全哈希函数：$H_V(\cdot)$、$H(\cdot)$、$H_{256}(\cdot)$；

[0059] 其中，q为大素数，F_q为包含q个元素的有限域，a、b为F_q中的元素，用于定义F_q上的一条椭圆曲线E；G为椭圆曲线的一个基点，其阶为素数；n为基点G的阶；

[0060] 系统密钥生成模块，用于由密钥生成中心KGC生成系统主公私钥(P_{pub},P_{pri})，其中，P_{pub}为系统主公钥，P_{pri}为系统主私钥；

[0061] 用户密钥提取模块，用于产生用户User的签名私钥，具体过程如下：

[0062] 1)用户User将身份标识ID发送给KGC，请求签名私钥；

[0063] 2)KGC收到私钥请求后，首先利用已有的身份认证方法确认ID与User身份一致，随

后，KGC在集合{1,2,…,n-1}中随机选取整数l并通过以下公式计算签名私钥的第一部分L；

[0064] $L=[l]G=(x_1,y_1)$

[0065] 其中，(x_1,y_1)表示L的横纵坐标。

[0066] 3)KGC根据L，通过以下公式计算签名私钥的第二部分α；

[0067] $h=H(ID||L)$

[0068] $\alpha=l+xh \bmod n$

[0069] 其中，H(•)表示安全哈希函数，符号||表示连接，modn表示模n运算；

[0070] 4)KGC将私钥(L,α)通过安全信道发送给用户User；

[0071] 5)用户User接收并秘密保存KGC发送的私钥(L,α)；

[0072] 用户签名模块，用于用户User产生消息M的数字签名(L,r,s)；

[0073] 签名验证模块，用于验证消息M'的签名(L',r',s')合法性。

[0074] 按上述方案，所述系统密钥生成模块中，由密钥生成中心KGC生成系统主公私钥的具体过程如下：

[0075] 步骤1)KGC从集合{1,2,…,n-1}中随机选取整数x作为主私钥$P_{pri}=x$，其中n表示SM2密码运算所使用的椭圆曲线点群的阶。

[0076] 步骤2)KGC根据所选主私钥x，通过以下公式计算并公布系统主公钥P_{pub}，

[0077] $P_{pub}=[x]G=(x_G,y_G)$

[0078] 其中，G为SM2数字签名系统参数中椭圆曲线点群的基点，[x]G表示点乘运算，(x_G,y_G)表示公钥的横纵坐标。

[0079] 按上述方案，所述用户签名模块中的签名过程具体如下：

[0080] 步骤1)用户User根据以下公式计算Z_A并置$\overline{M}=Z_A\|M$

[0081] $Z_A=H_{256}(ENTLA||ID||a||b||x_G||y_G||x_1||y_1)$

[0082] 其中，H_{256}表示安全哈希函数，a,b为椭圆曲线参数，ENTLA表示当前用户ID长度。

[0083] 步骤2)用户User根据以下公式计算e并将e转换为整数，

[0084] $e=H_v(\overline{M})$

[0085] 其中，H_V(•)表示安全哈希函数；

[0086] 步骤3)用户User在集合{1,2,…,n-1}中随机选择整数k，根据以下公式计算椭圆曲线点K，

[0087] $K=[k]G=(x_k,y_k)$

[0088] 并将x_k转换为整数类型；

[0089] 步骤4)用户User根据以下公式计算部分签名r并判断r，如果r=0或r+k=n成立，则返回步骤3)，否则执行步骤5)；

[0090] $r=(e+x_k) \bmod n$

[0091] 步骤5)用户User根据以下公式计算部分签名s并判断s，如果s=0成立，则返回步骤3)，否则执行步骤6)；

[0092] $s=(1+\alpha)^{-1}(k-r\alpha) \bmod n$

[0093] 其中，$(1+\alpha)^{-1}$为(1+α)的模n乘法逆；

[0094] 步骤6)用户User输出消息M签名为(L,r,s)。

[0095] 按上述方案，所述签名验证模块中验证消息M'的签名(L',r',s')合法性的具体验证过程如下：

[0096] 步骤1)检查r'是否属于集合{1,2,…,n-1}中，如果不是则验证不通过；否则，检查s'是否属于{1,2,…,n-1}，如果不是则验证不通过，否则进入步骤2)；

[0097] 步骤2)根据以下公式计算Z'_A并置$\overline{M'}=Z'_A\|M'$；

[0098] $Z'_A=H_{256}(ENTLA'||ID||a||b||x_G||y_G||x_1||y_1)$

[0099] 步骤3)根据以下公式计算e'，并将e'转换为整数；

[0100] $e'=H_v(\overline{M'})$

[0101] 步骤4)根据以下公式计算并判断t，如果t＝0成立，则消息签名为非法，结束验证过程，否则进入步骤5)；

[0102] $t=r'+s' \bmod n$

[0103] 步骤5)根据以下公式先后计算h'和椭圆曲线点K'，

[0104] $h'=H(ID||L')$

[0105] $K'=(x'_k,y'_k)=s'G+t(L'+h'P_{pub})$

[0106] 步骤6)根据以下公式计算并判断R'，如果R'＝r'成立，则消息签名合法，否则消息签名非法；

[0107] $R'=(e'+x'_k) \bmod n$。

[0108] 本发明产生的有益效果是：本发明在SM2签名算法整体架构不改变的基础上，使用了新型的用户私钥的产生方式，避免公钥证书的管理和维护。本发明未引入双线性对等耗时操作，在实现中具有较高的效率。因此，本发明设计的数字签名方案具有强安全性、高效率性、低开销性等特点。

附图说明

[0109] 下面将结合附图及实施例对本发明作进一步说明，附图中：

[0110] 图1是本发明实施例的方法流程图；

[0111] 图2是本发明实施例的系统主密钥生成实例流程示意图。

[0112] 图3是本发明实施例的用户密钥提取实例流程示意图。

[0113] 图4是本发明实施例的用户签名实例流程示意图。

[0114] 图5是本发明实施例的签名验证实例流程示意图。

具体实施方式

[0115] 为了使本发明的目的、技术方案及优点更加清楚明白，以下结合实施例，对本发明进行进一步详细说明。应当理解，此处所描述的具体实施例仅用以解释本发明，并不用于限定本发明。

[0116] 一.本发明实施例中的符号及定义

[0117] KGC:密钥生成中心。

[0118] A,B:用户。

[0119] ID_A:用户的身份。

[0120] ENTLA:ID_A的比特长度转换而成的两个字节。

[0121] q:大素数。

[0122] F_q:包含q个元素的有限域。

[0123] a,b:F_q中的元素,它们定义F_q上的一条椭圆曲线E。

[0124] $E(F_q)$:F_q上椭圆曲线E的所有有理点(包括无穷远点O)组成的集合。

[0125] $\#E(F_q)$:$E(F_q)$上点的数目,称为椭圆曲线$E(F_q)$的阶。

[0126] O:椭圆曲线上的一个特殊点,称为无穷远点或零点。

[0127] G:椭圆曲线的一个基点,其阶为素数。

[0128] [u]G:椭圆曲线的一个基点G的u倍点。即,[u]G=G+G+…+G,u是正整数。n:基点G的阶(n是$\#E(F_q)$的素因子)

[0129] x_G,y_G,x_k,y_k,x_1,y_1:F_q中的元素。

[0130] x,l,k:从{1,2,…,n-1}中选取的随机数。

[0131] $H_v(\cdot)$:消息摘要长度为V比特的密码杂凑函数。

[0132] $H(\cdot),H_{256}(\cdot)$:安全的密码杂凑函数。

[0133] P_{ri}:P_{ri}=x系统的签名主私钥。

[0134] P_{pub}:$P_{pub}=[x]G=(x_G,y_G)$,系统的签名主公钥。

[0135] M:待签名的消息。

[0136] M':待验证的消息。

[0137] e:密码杂凑函数作用于消息M的输出值。

[0138] e':密码杂凑函数作用于消息M'的输出值。

[0139] x||y:x与y的拼接,其中x和y是比特串或字节串。

[0140] Z_A:用户的可辩别标识,部分椭圆曲线系统参数和用户选定参数的杂凑值。

[0141] (L,r,s):发送的签名。

[0142] (L',r',s'):收到的签名。

[0143] 如图1所示,一种基于SM2的身份基的数字签名方法,具体步骤如下:

[0144] 步骤1.初始化(Setup):该步骤主要用于产生整个签名系统所需参数。参数包括:椭圆曲线相关参数(q,F_q,a,b,n,G)、安全哈希函数($H_v(\cdot),H(\cdot),H_{256}(\cdot)$)等。本发明是基于SM2数字签名算法的改进和优化,因此,与SM2使用相同系统参数,具体参数符号定义参见具体实施方式中(一.符号及定义)。

[0145] 步骤2.系统主密钥生成(KeyGen):如图2,该步骤主要用于产生系统主公私钥(P_{pub},P_{pri}),由密钥生成中心(KGC)执行,其中P_{pub}为系统主公钥和P_{pri}为系统主私钥。具体过程如下:

[0146] 步骤2.1:KGC从集合{1,2,…,n-1}中随机选取整数x作为主私钥P_{pri}=x。

[0147] 步骤2.2:KGC根据所选主私钥x,通过以下公式计算并公布系统主公钥P_{pub},

[0148] $P_{pub}=[x]G=(x_G,y_G)$

[0149] 步骤3.用户密钥提取(Extract):如图3,该步骤主要用于产生用户A的签名私钥。该步骤为本发明主要创新点。具体过程如下:

[0150] 步骤3.1:用户A将身份标识ID_A发送给KGC,请求签名私钥。

[0151] 步骤3.2:KGC收到私钥请求后,首先利用已有的身份认证方法确认ID_A与用户A身

份一致。随后，KGC在集合{1,2,…,n-1}中随机选取整数l并通过以下公式计算签名私钥的第一部分L。

[0152] $L=[l]G=(x_l,y_l)$

[0153] 步骤3.3:KGC根据L,通过以下公式计算签名私钥的第二部分α

[0154] $h=H(ID||L)$

[0155] $\alpha=l+xh \bmod n$

[0156] 步骤3.4:KGC将私钥(L,α)通过安全信道发送给用户A。

[0157] 步骤3.5:用户A接收并秘密保存KGC发送的私钥(L,α)。

[0158] 步骤4.用户签名(Sign):如图4,该步骤主要用于用户A产生消息M数字签名(L,r,s)。该步骤与SM2签名算法中签名过程基本相同。具体过程如下:

[0159] 步骤4.1:用户A根据以下公式计算Z_A并置$\overline{M}=Z_A\|M$

[0160] $Z_A=H_{256}(ENTLA||ID||a||b||x_G||y_G||x_1||y_1)$

[0161] 步骤4.2:用户A根据以下公式计算e并将e转换为整数

[0162] $e=H_v(\overline{M})$

[0163] 步骤4.3:用户A在集合{1,2,…,n-1}中随机选择整数k,根据以下公式计算椭圆曲线点K,

[0164] $K=[k]G=(x_k,y_k)$

[0165] 并将x_k转换为整数类型。

[0166] 步骤4.4:用户A根据以下公式计算部分签名r并判断r,如果r=0或r+k=n成立,则返回步骤4.3,否则执行步骤4.5

[0167] $r=(e+x_k) \bmod n$

[0168] 步骤4.5:用户A根据以下公式计算部分签名s并判断s,如果s=0成立,则返回步骤4.3,否则执行步骤4.6

[0169] $s=(1+\alpha)^{-1}(k-r\alpha) \bmod n$

[0170] 步骤4.6:用户A输出消息M签名为(L,r,s)。

[0171] 上述过程与SM2数字签名中签名算法过程基本相同,如果将L视为用户公钥,α视为用户私钥,则算法过程与SM2_Sign()算法一致。

[0172] 步骤5.签名验证(Verify):如图5,该步骤主要用于用户B验证用户A关于消息M'的签名(L',r',s')合法性。

[0173] 因为接收到的签名不一定是合法签名,所以才需要对签名进行验证,此处加了撇用以区分真正的签名和通过网络接收的签名。

[0174] 具体验证过程如下:

[0175] 步骤5.1:用户B检查r'是否属于集合{1,2,…,n-1}中,如果不是则验证不通过;否则,检查s'是否属于{1,2,…,n-1},如果不是则验证不通过,否则进入步骤5.2。

[0176] 步骤5.2:用户B根据以下公式计算Z'$_A$并置$\overline{M'}=Z_A'\|M'$

[0177] $Z'_A=H_{256}(ENTLA'||ID||a||b||x_G||y_G||x_1||y_1)$

[0178] 步骤5.3:用户B根据以下公式计算e'并将e'转换为整数,

[0179] $e'=H_v(\overline{M'})$

[0180] 步骤5.4:用户B根据以下公式计算并判断t,如果t=0成立,则消息签名为非法,结束验证过程,否则进入步骤5.5

[0181] $t=r'+s' \bmod n$

[0182] 步骤5.5:用户B根据以下公式先后计算h'和椭圆曲线点K':

[0183] $h'=H(ID||L')$

[0184] $K'=(x'_k,y'_k)=s'G+t(L'+h'P_{pub})$

[0185] 步骤5.6:根据以下公式计算并判断R',如果R'=r'成立,则消息签名合法,否则消息签名非法。

[0186] $R'=(e'+x'_k) \bmod n$

[0187] 根据上述方法,本发明还提供一种基于SM2的身份基的数字签名系统,其特征在于,包括:

[0188] 系统初始化模块,用于产生整个签名系统所需参数,参数包括:椭圆曲线相关参数:q、F_q、a、b、n、G,和安全哈希函数:$H_v(\cdot)$、$H(\cdot)$、$H_{256}(\cdot)$;

[0189] 其中,q为大素数,F_q为包含q个元素的有限域,a、b为F_q中的元素,用于定义F_q上的一条椭圆曲线E;G为椭圆曲线的一个基点,其阶为素数;n为基点G的阶;

[0190] 系统密钥生成模块,用于由密钥生成中心KGC生成系统主公私钥(P_{pub},P_{pri}),其中,P_{pub}为系统主公钥,P_{pri}为系统主私钥;

[0191] 具体过程如下:

[0192] 步骤1)KGC从集合{1,2,…,n-1}中随机选取整数x作为主私钥$P_{pri}=x$,其中n表示SM2密码运算所使用的椭圆曲线点群的阶。

[0193] 步骤2)KGC根据所选主私钥x,通过以下公式计算并公布系统主公钥P_{pub},

[0194] $P_{pub}=[x]G=(x_G,y_G)$

[0195] 其中,G为SM2数字签名系统参数中椭圆曲线点群的基点,[x]G表示点乘运算,(x_G,y_G)表示公钥的横纵坐标。

[0196] 用户密钥提取模块,用于产生用户User的签名私钥,具体过程如下:

[0197] 1)用户User将身份标识ID发送给KGC,请求签名私钥;

[0198] 2)KGC收到私钥请求后,首先利用已有的身份认证方法确认ID与User身份一致,随后,KGC在集合{1,2,…,n-1}中随机选取整数l并通过以下公式计算签名私钥的第一部分L;

[0199] $L=[l]G=(x_l,y_l)$

[0200] 其中,(x_l,y_l)表示L的横纵坐标。

[0201] 3)KGC根据L,通过以下公式计算签名私钥的第二部分α;

[0202] $h=H(ID||L)$

[0203] $\alpha=l+xh \bmod n$

[0204] 其中,$H(\cdot)$表示安全哈希函数,符号||表示连接,modn表示模n运算;

[0205] 4)KGC将私钥(L,α)通过安全信道发送给用户User;

[0206] 5)用户User接收并秘密保存KGC发送的私钥(L,α);

[0207] 用户签名模块,用于用户User产生消息M的数字签名(L,r,s);

[0208] 用户签名模块中的签名过程具体如下:

[0209] 步骤1)用户User根据以下公式计算Z_A并置$\overline{M}=Z_A\|M$

[0210] $Z_A=H_{256}(ENTLA||ID||a||b||x_G||y_G||x_1||y_1)$

[0211] 其中，H_{256}表示安全哈希函数，a，b为椭圆曲线参数，ENTLA表示当前用户ID长度。

[0212] 步骤2)用户User根据以下公式计算e并将e转换为整数，

[0213] $e=H_v(\overline{M})$

[0214] 其中，$H_V(\bullet)$表示安全哈希函数；

[0215] 步骤3)用户User在集合{1,2,…,n-1}中随机选择整数k，根据以下公式计算椭圆曲线点K，

[0216] $K=[k]G=(x_k,y_k)$

[0217] 并将x_k转换为整数类型；

[0218] 步骤4)用户User根据以下公式计算部分签名r并判断r，如果r＝0或r+k＝n成立，则返回步骤3)，否则执行步骤5)；

[0219] $r=(e+x_k)\ mod n$

[0220] 步骤5)用户User根据以下公式计算部分签名s并判断s，如果s＝0成立，则返回步骤3)，否则执行步骤6)；

[0221] $s=(1+\alpha)^{-1}(k-r\alpha)\ mod n$

[0222] 其中，$(1+\alpha)^{-1}$为$(1+\alpha)$的模n乘法逆；

[0223] 步骤6)用户User输出消息M签名为(L,r,s)。

[0224] 签名验证模块，用于验证消息M'的签名(L',r',s')合法性。具体验证过程如下：

[0225] 步骤1)检查r'是否属于集合{1,2,…,n-1}中，如果不是则验证不通过；否则，检查s'是否属于{1,2,…,n-1}，如果不是则验证不通过，否则进入步骤2)；

[0226] 步骤2)根据以下公式计算Z'A并置$\overline{M'}=Z_A'\|M'$；

[0227] $Z'_A=H_{256}(ENTLA'||ID||a||b||x_G||y_G||x_1||y_1)$

[0228] 步骤3)根据以下公式计算e'，并将e'转换为整数；

[0229] $e'=H_v(\overline{M'})$

[0230] 步骤4)根据以下公式计算并判断t，如果t＝0成立，则消息签名为非法，结束验证过程，否则进入步骤5)；

[0231] $t=r'+s'\ mod n$

[0232] 步骤5)根据以下公式先后计算h'和椭圆曲线点K'，

[0233] $h'=H(ID||L')$

[0234] $K'=(x'_k,y'_k)=s'G+t(L'+h'P_{pub})$

[0235] 步骤6)根据以下公式计算并判断R'，如果R'＝r'成立，则消息签名合法，否则消息签名非法；

[0236] $R'=(e'+x'_k)\ mod n$。

[0237] 应当理解的是，对本领域普通技术人员来说，可以根据上述说明加以改进或变换，而所有这些改进和变换都应属于本发明所附权利要求的保护范围。

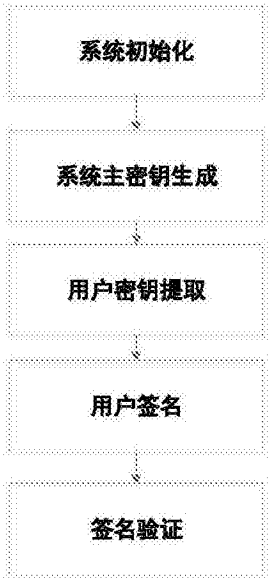

系统初始化
系统主密钥生成
用户密钥提取
用户签名
签名验证

图1

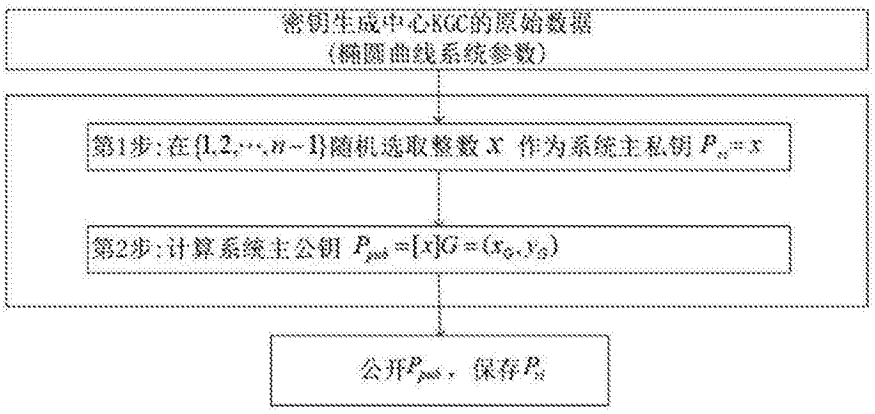

密钥生成中心KGC的原始数据
(椭圆曲线系统参数)
第1步：在{1,2,…,n-1}随机选取整数 x 作为系统主私钥 $P_{pr}=x$
第2步：计算系统主公钥 $P_{pub}=[x]G=(x_G,y_G)$
公开P_{pub}，保存P_{pr}

图2

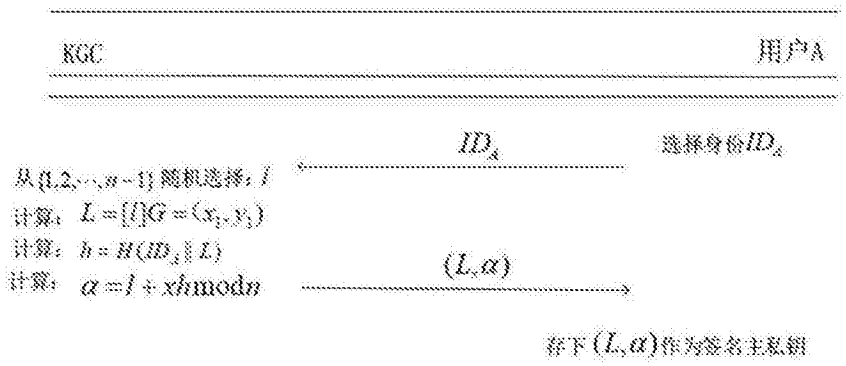

KGC
用户A
ID_A
选择身份ID_A
从{1,2,…,n-1}随机选择：l
计算：$L=[l]G=(x_1,y_1)$
计算：$h=H(ID_A\|L)$
计算：$\alpha=l+xh \bmod n$
(L,α)
存下(L,α)作为签名主私钥

图3

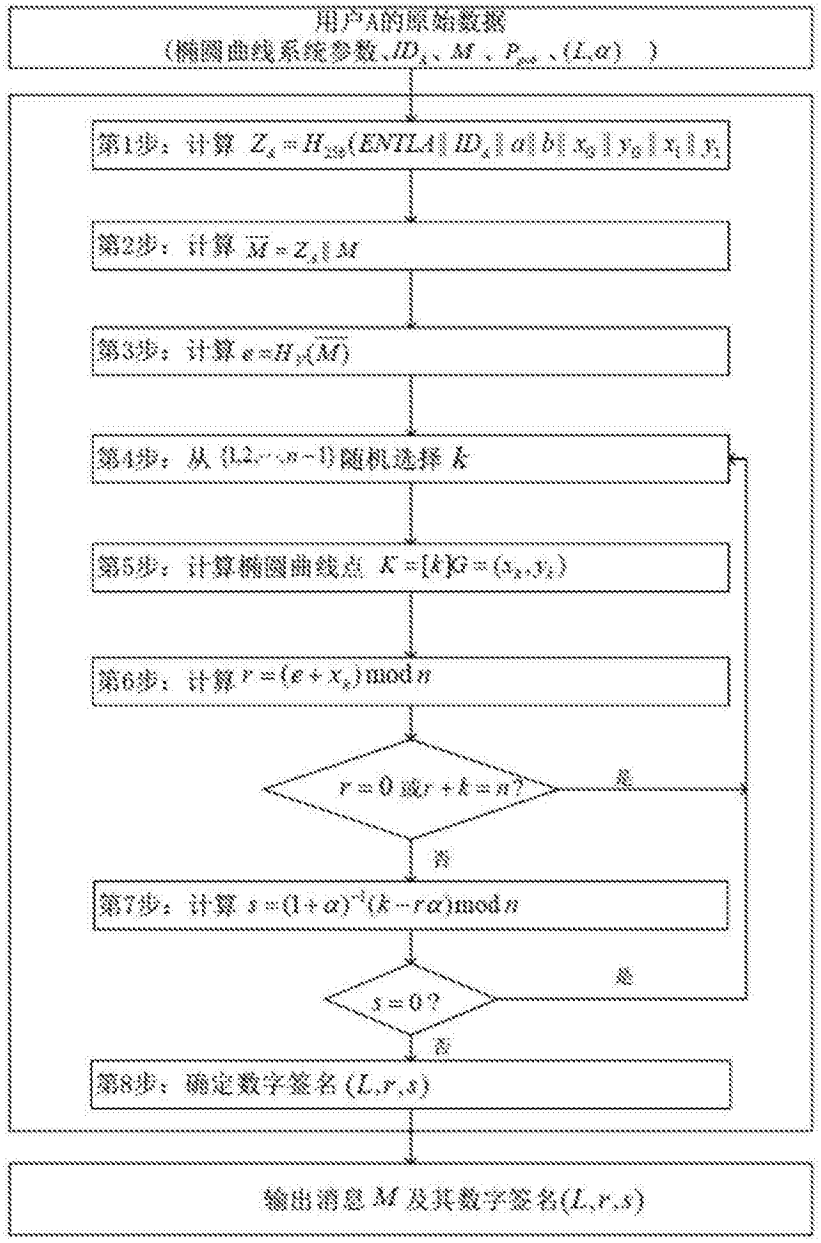

用户A的原始数据
（椭圆曲线系统参数、ID_A、M、P_{pub}、(L,α)）
第1步：计算 $Z_A=H_{256}(ENTLA\|ID_A\|a\|b\|x_G\|y_G\|x_A\|y_A$
第2步：计算 $\overline{M}=Z_A\|M$
第3步：计算 $e=H_v(\overline{M})$
第4步：从 $[1,2,\cdots,n-1]$ 随机选择 k
第5步：计算椭圆曲线点 $K=[k]G=(x_1,y_1)$
第6步：计算 $r=(e+x_1)\bmod n$
$r=0$ 或 $r+k=n$？
是
否
第7步：计算 $s=(1+\alpha)^{-1}(k-r\alpha)\bmod n$
$s=0$？
是
否
第8步：确定数字签名 (L,r,s)
输出消息 M 及其数字签名(L,r,s)

图4

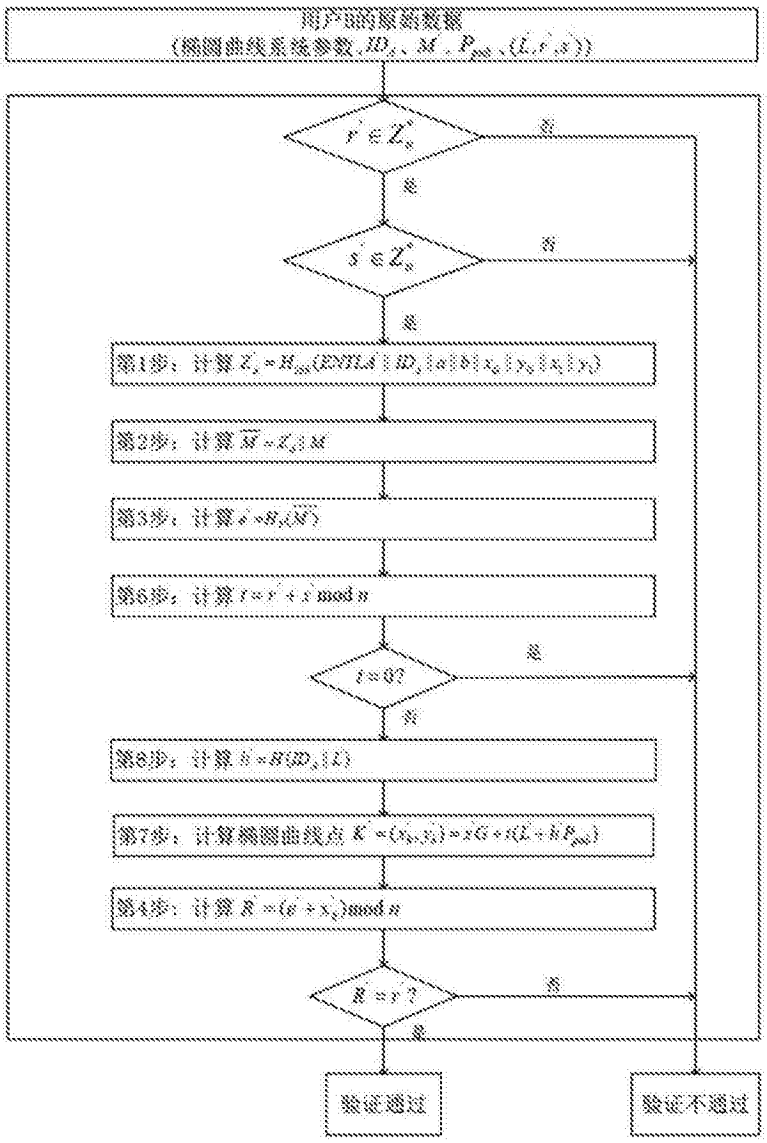

用户B的原始数据
(椭圆曲线系统参数, ID_A、M、P_{pub}、(L, r', s'))
$r' \in Z_n^*$
否
是
$s' \in Z_n^*$
否
是
第1步：计算 $Z_A = H_{256}(ENTL_A \| ID_A \| a \| b \| x_G \| y_G \| x_A \| y_A)$
第2步：计算 $\overline{M'} = Z_A \| M$
第3步：计算 $e' = H_v(\overline{M'})$
第4步：计算 $t = r' + s' \bmod n$
$t = 0$?
是
否
第5步：计算 $h = H(ID_A \| L)$
第7步：计算椭圆曲线点 $R' = (x_1', y_1') = sG + t(L + hP_{pub})$
第4步：计算 $R' = (e' + x_1') \bmod n$
$R' = r'$?
否
是
验证通过
验证不通过

图5