ICS 35.040
L 80
备案号：

中华人民共和国密码行业标准

GM/T XXXX-XXXX

SM9 标识密码算法
第 2 部分：数字签名算法

Identity-Based Cryptographic Algorithms SM9
Part 2: Digital Signature Algorithm

（报批稿）

（在提交反馈意见时，请将您知道的相关专利连同支持性文件一并附上）

××××-××-××发布　　××××-××-××实施

国家密码管理局　发布

目　　次

前　言

本标准依据 GB/T 1.1-2009 给出的规则起草。

GM/T ××××—××××《SM9 标识密码算法》分为五个部分：

——第 1 部分：总则

——第 2 部分：数字签名算法

——第 3 部分：密钥交换协议

——第 4 部分：密钥封装机制和公钥加密算法

——第 5 部分：参数定义

本部分为 GM/T ××××—××××的第 2 部分。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

本部分由密码行业标准化技术委员会提出并归口。

本部分起草单位：国家信息安全工程技术研究中心、深圳奥联科技有限公司、武汉大学、上海交通大学、中科院信息工程研究所、北方信息技术研究所。

本部分主要起草人：陈晓、程朝辉、叶顶峰、胡磊、陈建华、路贝可、季庆光、曹珍富、袁文恭、刘平、马宁、李增欣、王学进、袁峰、杨恒亮、张青坡、马艳丽、浦雨三、唐英、孙移盛、安萱。

引　言

A.Shamir 在1984年提出了标识密码(Identity-Based Cryptography)的概念，在标识密码系统中，用户的私钥由密钥生成中心(KGC)根据主密钥和用户标识计算得出，用户的公钥由用户标识唯一确定，从而用户不需要通过第三方保证其公钥的真实性。与基于证书的公钥密码系统相比，标识密码系统中的密钥管理环节可以得到适当简化。

1999年，K.Ohgishi、R.Sakai 和M.Kasahara在日本提出了用椭圆曲线对(pairing)构造基于标识的密钥共享方案；2001年，D.Boneh和M.Franklin，以及R.Sakai、K.Ohgishi和M.Kasahara 等人独立提出了用椭圆曲线对构造标识公钥加密算法。这些工作引发了标识密码的新发展，出现了一批用椭圆曲线对实现的标识密码算法，其中包括数字签名算法、密钥交换协议、密钥封装机制和公钥加密算法等。

椭圆曲线对具有双线性的性质，它在椭圆曲线的循环子群与扩域的乘法循环子群之间建立联系，构成了双线性DH、双线性逆DH、判定性双线性逆DH、τ-双线性逆DH和τ-Gap-双线性逆DH等难题，当椭圆曲线离散对数问题和扩域离散对数问题的求解难度相当时，可用椭圆曲线对构造出安全性和实现效率兼顾的标识密码。

本部分描述了用椭圆曲线对实现的基于标识的数字签名算法。

SM9 标识密码算法
第 2 部分：数字签名算法

1 范围

GM/T ××××—××××的本部分规定了用椭圆曲线对实现的基于标识的数字签名算法，包括数字签名生成算法和验证算法，并给出了数字签名与验证算法及其相应的流程。

本部分适用于接收者通过签名者的标识验证数据的完整性和数据发送者的身份，也适用于第三方确定签名及所签数据的真实性。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅所注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GM/T 0004—2012 SM3密码杂凑算法

GM/T ××××—×××× SM9标识密码算法 第1部分：总则

3 术语和定义

下列术语和定义适用于本部分。

3.1

消息 message

任意有限长度的比特串。

3.2

签名消息 signed message

由消息以及该消息的数字签名部分所组成的一组数据元素。

3.3

签名密钥 signature key

在数字签名生成过程中由签名者专用的秘密数据元素，即签名者的私钥。

3.4

签名主密钥 signature master key

处于标识密码密钥分层结构最顶层的密钥，包括签名主私钥和签名主公钥，其中签名主公钥公开，签名主私钥由 KGC 秘密保存。KGC 用签名主私钥和用户的标识生成用户的签名私钥。在标识密码

中，签名主私钥一般由 KGC 通过随机数发生器产生，签名主公钥由签名主私钥结合系统参数产生。

3.5

标识 identity

可唯一确定一个实体身份的信息。标识应由实体无法否认的信息组成，如实体的可识别名称、电子邮箱、身份证号、电话号码、街道地址等。

3.6

密钥生成中心 key generation center (KGC)

在本部分中，负责选择系统参数、生成签名主密钥并产生用户签名私钥的可信机构。

4 符号

下列符号适用于本部分。

A, B：使用标识密码系统的两个用户。

cf：椭圆曲线阶相对于 N 的余因子。

cid：用一个字节表示的曲线的识别符，其中 0x10 表示 F_p(素数 $p>2^{191}$) 上常曲线(即非超奇异曲线)，0x11 表示 F_p 上超奇异曲线，0x12 表示 F_p 上常曲线及其扭曲线。

ds_A：用户 A 的签名私钥。

e：从 $\mathbb{G}_1\times\mathbb{G}_2$ 到 $\mathbb{G}_T$ 的双线性对。

eid：用一个字节表示的双线性对 e 的识别符，其中 0x01 表示 Tate 对，0x02 表示 Weil 对，0x03 表示 Ate 对，0x04 表示 R-Ate 对。

$\mathbb{G}_T$：阶为素数N的乘法循环群。

$\mathbb{G}_1$：阶为素数N的加法循环群。

$\mathbb{G}_2$：阶为素数N的加法循环群。

g^u：乘法群 $\mathbb{G}_T$ 中元素 g 的 u 次幂，即 $g^u=\underbrace{g\cdot g\cdot\ldots\cdot g}_{u个}$，$u$ 是正整数。

$H_v(\ )$：密码杂凑函数。

$H_1(\ ), H_2(\ )$：由密码杂凑函数派生的密码函数。

hid：在本部分中，用一个字节表示的签名私钥生成函数识别符，由 KGC 选择并公开。

(h, S)：发送的签名。

(h', S')：收到的签名。

ID_A：用户 A 的标识，可以唯一确定用户 A 的公钥。

M：待签名消息。

M'：待验证消息。

mod n：模 n 运算。例如，23 mod 7=2。

N：循环群$\mathbb{G}_1$、$\mathbb{G}_2$和$\mathbb{G}_T$的阶，为大于2^{191}的素数。

$P_{pub\text{-}s}$：签名主公钥。

P_1：群$\mathbb{G}_1$的生成元。

P_2：群$\mathbb{G}_2$的生成元。

ks：签名主私钥。

$<P>$：由元素 P 生成的循环群。

$[u]P$：加法群 $\mathbb{G}_1$、$\mathbb{G}_2$ 中元素 P 的 u 倍。

$\lceil x \rceil$：顶函数，不小于 x 的最小整数。例如，$\lceil 7 \rceil = 7, \lceil 8.3 \rceil = 9$。

$\lfloor x \rfloor$：底函数，不大于 x 的最大整数。例如，$\lfloor 7 \rfloor = 7, \lfloor 8.3 \rfloor = 8$。

$x \| y$：x 与 y 的拼接，x 和 y 是比特串或字节串。

$[x, y]$：不小于 x 且不大于 y 的整数的集合。

5 算法参数与辅助函数

5.1 总则

本部分规定了一个用椭圆曲线对实现的基于标识的数字签名算法。该算法的签名者持有一个标识和一个相应的签名私钥，该签名私钥由密钥生成中心通过签名主私钥和签名者的标识结合产生。签名者用自身签名私钥对数据产生数字签名，验证者用签名者的标识验证签名的可靠性。

在签名的生成和验证过程之前，都要用密码杂凑函数对待签消息M和待验证消息M'进行压缩。

5.2 系统参数组

系统参数组包括曲线识别符 cid；椭圆曲线基域 F_q 的参数；椭圆曲线方程参数 a 和 b；扭曲线参数 β(若 cid 的低 4 位为 2)；曲线阶的素因子 N 和相对于 N 的余因子 cf；曲线 $E(F_q)$相对于 N 的嵌入次数 k；$E(F_{q^{d_1}})$(d_1 整除 k)的 N 阶循环子群 $\mathbb{G}_1$ 的生成元 P_1；$E(F_{q^{d_2}})$ (d_2 整除 k)的 N 阶循环子群 $\mathbb{G}_2$ 的生成元 P_2；双线性对 e 的识别符 eid；（选项）$\mathbb{G}_2$ 到 $\mathbb{G}_1$ 的同态映射 ψ。

双线性对 e 的值域为 N 阶乘法循环群 $\mathbb{G}_T$。

系统参数的详细描述及其验证参见GM/T ××××—××××第1部分第7章。

5.3 系统签名主密钥和用户签名密钥的产生

KGC产生随机数$ks \in [1, N-1]$作为签名主私钥，计算$\mathbb{G}_2$中的元素$P_{pub\text{-}s}=[ks]P_2$作为签名主公钥，则签名主密钥对为$(ks, P_{pub\text{-}s})$。KGC秘密保存ks，公开$P_{pub\text{-}s}$。

KGC选择并公开用一个字节表示的签名私钥生成函数识别符hid。

用户A的标识为ID_A，为产生用户A的签名私钥ds_A，KGC首先在有限域F_N上计算$t_1=H_1(ID_A \| hid, N)+ks$，若$t_1=0$则需重新产生签名主私钥，计算和公开签名主公钥，并更新已有用户的签名私钥；否则计算$t_2 = ks \cdot t_1^{-1}$，然后计算$ds_A=[t_2]P_1$。

5.4 辅助函数

5.4.1 概述

在本部分规定的基于标识的数字签名算法中，涉及到两类辅助函数：密码杂凑函数与随机数发生器。

5.4.2 密码杂凑函数

5.4.2.1 密码杂凑函数 $H_v(\ )$

密码杂凑函数$H_v(\ )$的输出是长度恰为v比特的杂凑值。本部分规定使用国家密码管理局批准的密码杂凑函数，如SM3密码杂凑算法。

5.4.2.2 密码函数 $H_1(\ )$

密码函数$H_1(Z, n)$的输入为比特串Z和整数n，输出为一个整数$h_1 \in [1, n-1]$。 $H_1(Z, n)$需要调用密码杂凑函数$H_v(\ )$。关于密码杂凑函数$H_v(\ )$，应符合本部分5.4.2.1的规定。

密码函数 $H_1(Z, n)$：

输入：比特串 Z，整数 n。

输出：整数$h_1 \in [1, n-1]$。

步骤 1：初始化一个 32 比特构成的计数器 ct=0x00000001；

步骤 2：计算 $hlen=8\times\lceil(5\times(\log_2 n))/32\rceil$；

步骤 3：对 i 从 1 到$\lceil hlen/v\rceil$执行：

步骤 3.1：计算 $Ha_i=H_v(0x01\|Z\|ct)$；

步骤 3.2：ct++；

步骤 4：若 $hlen/v$ 是整数，令 $Ha!_{\lceil hlen/v\rceil} = Ha_{\lceil hlen/v\rceil}$，

否则令 $Ha!_{\lceil hlen/v\rceil}$为 $Ha_{\lceil hlen/v\rceil}$最左边的$(hlen-(v\times\lfloor hlen/v\rfloor))$比特；

步骤 5：令 $Ha = Ha_1 \| Ha_2 \| \cdots \| Ha_{\lceil hlen/v\rceil-1} \| Ha!_{\lceil hlen/v\rceil}$，按本标准第 1 部分 6.2.4 和 6.2.3 给出的细节将 Ha 的数据类型转换为整数；

步骤6：计算$h_1=(Ha \bmod (n-1))+1$。

5.4.2.3 密码函数 $H_2(\ )$

密码函数$H_2(Z, n)$的输入为比特串Z和整数n，输出为一个整数$h_2 \in [1, n-1]$。$H_2(Z, n)$需要调用密码杂凑函数$H_v(\ )$。关于密码杂凑函数$H_v(\ )$，应符合本部分5.4.2.1的规定。

密码函数 $H_2(Z, n)$：

输入：比特串 Z，整数 n。

输出：整数$h_2 \in [1, n-1]$。

步骤 1：初始化一个 32 比特构成的计数器 ct=0x00000001；

步骤 2：计算 $hlen=8\times\lceil(5\times(\log_2 n))/32\rceil$；

步骤 3：对 i 从 1 到$\lceil hlen/v\rceil$执行：

步骤 3.1：计算 $Ha_i=H_v(0x02\|Z\|ct)$；

步骤 3.2：ct++；

步骤 4：若 $hlen/v$ 是整数，令 $Ha!_{\lceil hlen/v\rceil} = Ha_{\lceil hlen/v\rceil}$，

否则令 $Ha!_{\lceil hlen/v\rceil}$为 $Ha_{\lceil hlen/v\rceil}$最左边的$(hlen-(v\times\lfloor hlen/v\rfloor))$比特；

步骤 5：令 $Ha = Ha_1 \| Ha_2 \| \cdots \| Ha_{\lceil hlen/v\rceil-1} \| Ha!_{\lceil hlen/v\rceil}$，按本标准第 1 部分 6.2.4 和 6.2.3 给出的细节将 Ha 的数据类型转换为整数；

步骤6：计算$h_2=(Ha \bmod (n-1))+1$。

5.4.3 随机数发生器

本部分规定使用国家密码管理局批准的随机数发生器。

6 数字签名生成算法及流程

6.1 数字签名生成算法

设待签名的消息为比特串 M，为了获取消息 M 的数字签名(h, S)，作为签名者的用户 A 应实现以下运算步骤：

A1： 计算群 $\mathbb{G}_T$ 中的元素 $g = e(P_1, P_{pub\text{-}s})$；

A2： 产生随机数 $r \in [1, N-1]$；

A3： 计算群 G_T 中的元素 $w = g^r$，按 GM/T ××××—××××第 1 部分 6.2.6 和 6.2.5 给出的细节将 w 的数据类型转换为比特串；

A4： 计算整数 $h = H_2(M||w, N)$；

A5： 计算整数 $l = (r-h) \bmod N$，若 $l = 0$ 则返回 A2；

A6： 计算群 G_1 中的元素 $S = [l]ds_A$；

A7： 按 GM/T ××××—××××第 1 部分 6.2.2 给出的细节将 h 的数据类型转换为字节串，按 GM/T ××××—××××第 1 部分 6.2.8 给出的细节将 S 的数据类型转换为字节串，消息 M 的签名为(h, S)。

6.2 数字签名生成算法流程

数字签名生成算法流程如图1。

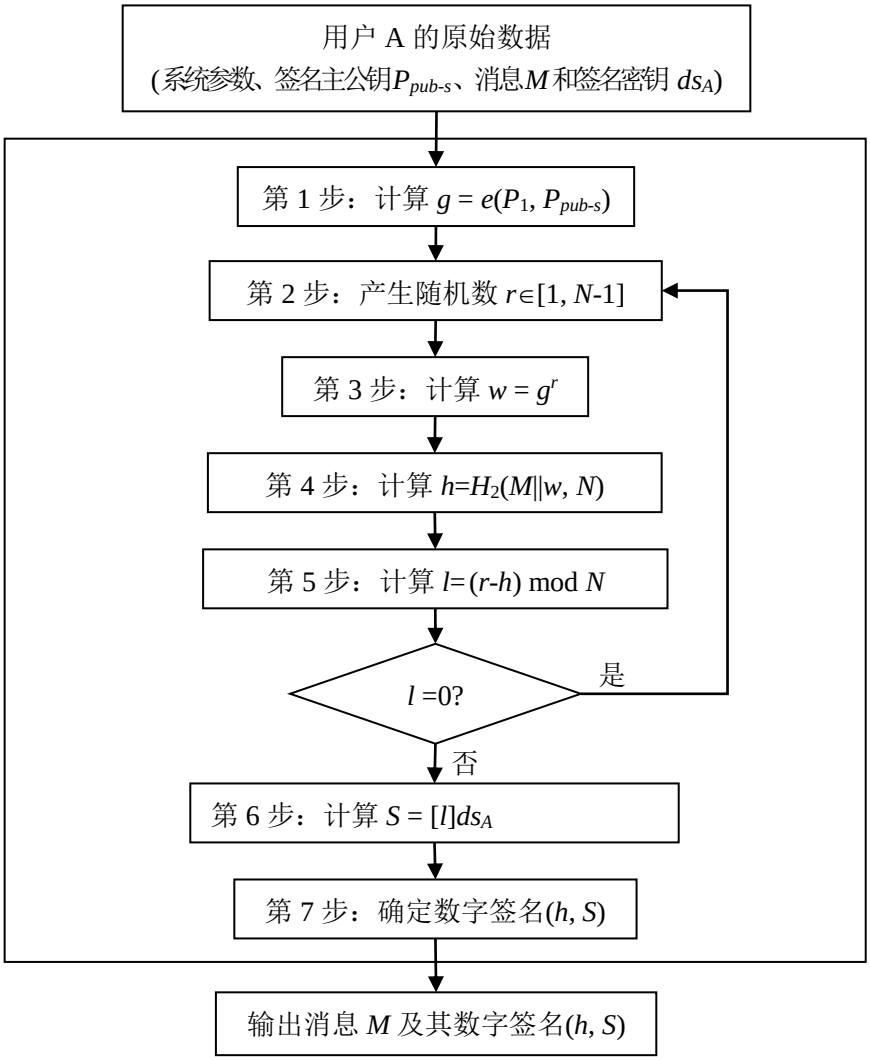

图1 数字签名生成算法流程

7 数字签名验证算法及流程

7.1 数字签名验证算法

为了检验收到的消息 M' 及其数字签名(h', S')，作为验证者的用户 B 应实现以下运算步骤：

B1： 按 GM/T ××××—××××第 1 部分 6.2.3 给出的细节将 h'的数据类型转换为整数，检验 $h' \in [1, N-1]$ 是否成立，若不成立则验证不通过；

B2： 按 GM/T ××××—××××第 1 部分 6.2.9 给出的细节将 S'的数据类型转换为椭圆曲线上的点，按 GM/T ××××—××××第 1 部分 4.5 给出的细节检验 $S' \in G_1$ 是否成立，若不成立则验证不通过；

B3： 计算群 G_T 中的元素 $g = e(P_1, P_{pub-s})$；

B4： 计算群 G_T 中的元素 $t = g^{h'}$；

B5： 计算整数 $h_1 = H_1(ID_A \| hid, N)$；

B6： 计算群 G_2 中的元素 $P = [h_1]P_2 + P_{pub-s}$；

B7： 计算群 G_T 中的元素 $u = e(S', P)$；

B8： 计算群 G_T 中的元素 $w' = u \cdot t$，按 GM/T ××××—××××第 1 部分 6.2.6 和 6.2.5 给出的细节将 w' 的数据类型转换为比特串；

B9： 计算整数 $h_2 = H_2(M' \| w', N)$，检验 $h_2 = h'$ 是否成立，若成立则验证通过；否则验证不通过。

7.2 数字签名验证算法流程

数字签名验证算法流程如图2。

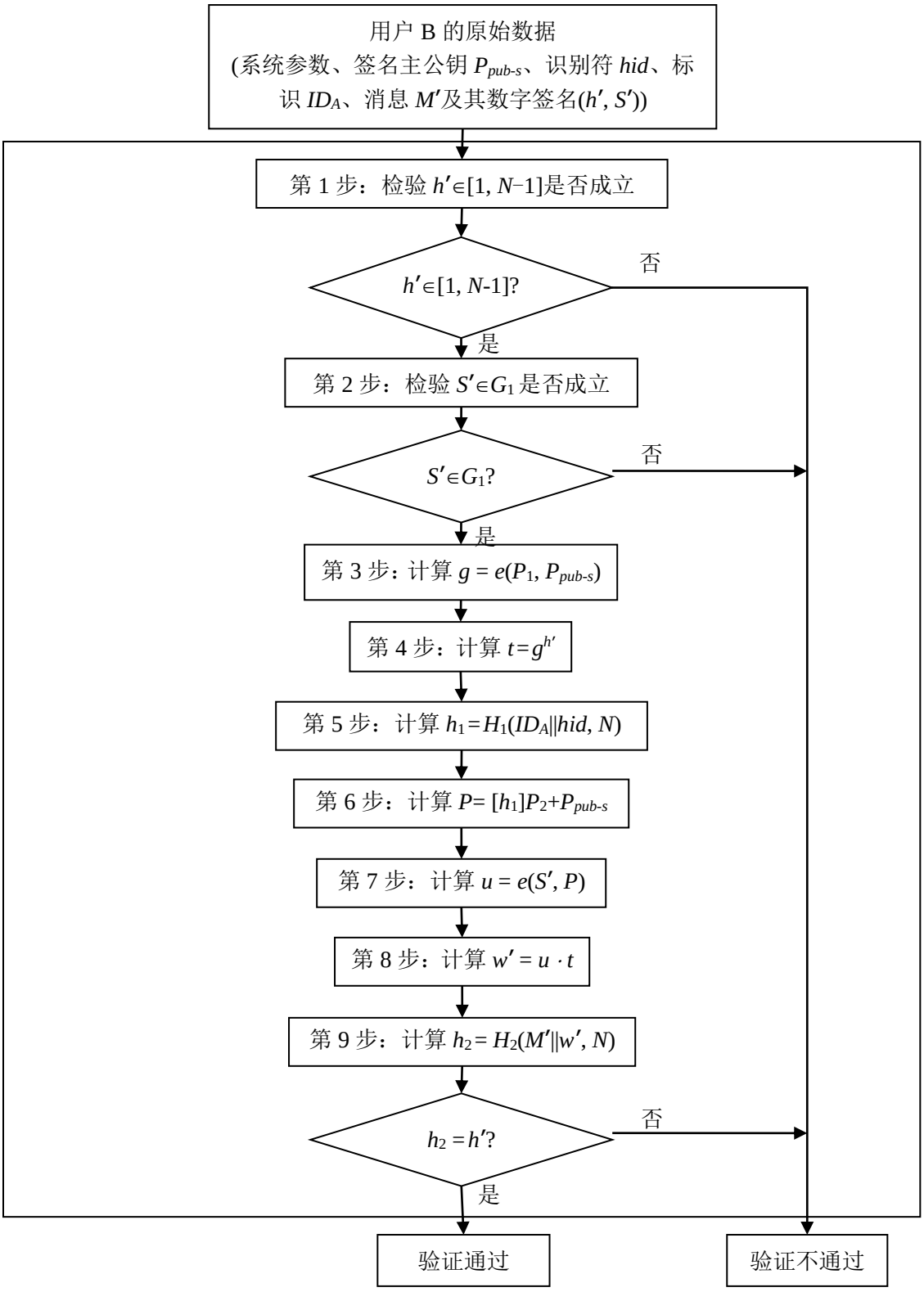

图2　数字签名验证算法流程