基于SM2的双方共同签名协议及其应用

苏吟雪 田海博

（中山大学数据科学与计算机学院广东省信息安全技术重点实验室 广州 510006）

摘 要 移动互联网近年来发展迅速，移动智能设备的持有率大大增加，使用范围也不断扩大，保护用户信息安全的重要性也随之提升，但由于设备的计算能力有限，增加了密钥泄露的威胁，移动设备中存储的敏感信息也日益成为攻击目标，导致在移动设备上生成的数字签名在司法举证时难以认定是私钥的所有人签署. 随着5G技术和物联网技术的发展，移动设备的应用将更加广泛，该问题亟待解决. 5G技术的特点包括高带宽和低延迟，这为该问题的解决提供了可能性. 双方共同签名是门限群签名的特殊形式，双方共同签名协议要求签名所用私钥的一部分存储在服务器中，增加了服务器认证用户的机会，进而加强了所生成数字签名的法律效力. SM2是国家密码管理局于2010年发布的椭圆曲线公钥密码算法，是国家公钥密码算法标准GM/T 0003.2—2012，包含了数字签名算法，密钥交换协议和公钥加密算法. 基于SM2的共同签名协议依旧较少，缺乏高效的可证明安全的共同签名协议. 因此本文提出了一个基于SM2的双方共同签名协议. 该协议适用于单个服务节点服务大量客户端的场景，例如5G环境下的物联网场景. 技术上看，该协议是可证明安全的，且服务器在进行一次共同签名时可以只进行一次标量乘计算. 在基本协议的基础上，考虑实际需求，我们给出了一个扩展的应用协议，增加了服务器对客户端的认证和数字证书颁发的流程.

关键词 SM2；双方共同签名；可证明安全；随机预言机

中图法分类号 TP391 **DOI号** 10.11897/SP.J.1016.2020.00701

A two-party SM2 signing Protocol and its application

SU Yin-Xue TIAN Hai-Bo

（*GuangDong Province Key Laboratory of Information Security Technology*, *School of Data and Computer Science*, *Sun Yat-Sen University*, *Guangzhou* 510006）

Abstract With the rapid development of mobile internet in recent years, the proportion of smartphone, tablet and other intelligent mobile devices is greatly increased and the using range of intelligent mobile devices is also expanding. On the one hand, the expansion of the scale of the intelligent mobile devices users has increased the importance of protecting user information security; on the other hand, due to the limitation of the computing ability of intelligent mobile devices, the threat of key disclosure which stored in those mobile devices increases. In other words, the sensitive information stored in mobile devices is increasingly becoming the attack target. As a result, it is difficult to judge the real signer of a digital signature which is generated on mobile devices at the time of judicial proof which reduces the legal validity of digital signature. With the deep integration and development of 5G technology and internet of things technology, the applications and using range of mobile devices would be more extensive, so the problem needs to be solved urgently. Fortunately, the typical characteristics of 5G technology include high bandwidth and low delay, which provides the possibility to solve the problem, such as improving the computing efficiency at the cost of

收稿日期：2019-08-19；在线出版日期：2020-02-07. 本课题得到得到国家重点研发计划（No. 2017YFB0802500），国家自然科学基金项目（No. 61672550，No. 619724290）和广东省自然科学基金项目（No. 2018A0303130133）资助. **苏吟雪**，硕士研究生，主要研究领域为信息安全、密码协议. E-mail：syxed@qq.com. **田海博**（通信作者），博士，副教授，主要研究领域为安全协议设计与分析、区块链技术. E-mail：tianhb@mail.sysu.edu.cn.

communication. The two-party signing protocol is a special case of threshold group signature. The two-party signing protocol requires that part of the private key used for signing is stored in the server, which increases the chance for the server to authenticate the user, and in turn strengthens the legal effect of the generated digital signature. SM2 is an elliptic curve public key cryptography algorithm released by the State Cryptography Administration in 2010 and its security mainly depends on the discrete logarithm problem of elliptic curve, and the SM2 algorithm includes digital signature algorithm, key exchange protocol and public key encryption algorithm. SM2 has become China's public key algorithm standard GM/T 0003.2-2012 and is of great significance to China's information security construction. Currently, there are still fewer protocols of two-party SM2 signing. The existing works are basically flawed in security provable or efficiency. Therefore, this paper proposes a new two-party signing protocol based on SM2 which could properly achieve the balance of the security provable and efficiency. This protocol is applicable to the scenario where a single service node serves a large number of clients, such as the internet of things scenario in 5G environment. The scheme is provable secure, and the server can perform only a scalar multiplication when it participates in a two-party signing, which is the advantage of the protocol. Meanwhile, there would be some new requirements in the actual operation of the common signature protocol, such as the signer need to issue a digital certificate for the common public key, or the one party want to authenticate the another party before using the private key part he stored. Therefore based on our algorithm and considering these practical requirements, we present an application protocol, which extends the procedures of client authentication and digital certificate generation.

Keywords SM2; two-party signing; provable secure; random oracle

1 引 言

移动互联网在近几年来发展十分迅速,智能手机,平板等等智能移动设备的持有和使用比例都大大增加,使用范围也不断扩大.据统计[1],截至2018年,我国网民规模达到8.29亿人,手机用户达到8.17亿人,网络购物用户规模达到6.10亿人,线下手机网络支付的用户已经到达5.83亿人.一方面,用户规模的扩大使保障用户信息安全的重要性上升;另一方面,由于智能移动设备本身的计算能力的限制,通常采用软件模块来保存密钥至本地或者智能芯片中[2],这增加了密钥泄露的威胁,甚至使得移动设备生成的数字签名不再具有法律效力.随着5G技术和物联网技术的深度融合发展,移动设备的数量只会越来越多,因此该问题越来越引起人们的重视.另一方面,5G技术的典型特点包括高带宽和低延迟,这为以通信代价换取计算效率的提升提供了基本条件.

双方共同签名是门限群签名[3]的特例.门限群签名是基于门限秘密共享[4]和数字签名技术的结合.根据是否有可信第三方参与[5],门限群签名可以大致分为两大类,这两类在设计上和应用上分别存在一些需要克服的难题.在(k, t)门限群签名中,k方共享密钥,任意大于等于t的共享方都可以对消息进行有效签名.从这个意义上,双方共同签名可以看成(2,2)门限群签名.在移动互联网场景下,双方共同签名主要有两个优点.第一,密钥的分散存储提升了私钥的安全性,提高了攻击门槛;第二,用户与服务器合作签名的方式也提供了服务器认证用户的机会,增强了用户所生成数字签名的法律效力.

SM2[6]是国家密码管理局于2010年发布的椭圆曲线公钥密码算法,包含了数字签名算法,密钥交换协议和公钥加密算法.SM2已成为我国公钥算法标准GM/T 0003.2—2012,并进入国际标准ISO/IEC 14888—3:2016中,对我国的信息安全建设有重要意义.椭圆曲线密码体制[7]是1985年提出的,它的安全性主要依赖于椭圆曲线离散对数困难问题.

当前,基于SM2的共同签名协议只有几件专利和少数几篇论文,缺乏高效的可证明安全的共同签

名协议. 本文的目标是给出一个新的基于SM2的双方共同签名协议,并给出实际的应用协议. 该协议主要有两个特点,一是可证明安全,二是服务器计算量较少. 应用协议包含了数字证书颁发和客户端认证的流程. 值得注意的是,该协议是针对SM2签名算法的特定结构设计的,并不是一般化的构造,不能直接推广到ECDSA等数字签名算法. 这与本文追求的效率和安全性目标一致.

1.1　相关工作

根据SM2签名是否有安全性证明,我们可以把基于SM2的共同签名协议分为两类.

Lin等人[8]在2014年公开了一个基于SM2的双方共同签名协议,签署双方只需要一轮通信和3次标量乘,效率极高. Jie等人[9]在2016年提出了一个无可信第三方的SM2门限签名协议. 该协议基于秘密共享和安全多方计算技术,各方协作生成共同签名的各个部分,直至获得最终签名,通信代价和计算代价都比较高. Zhang等人[10]在2017年公开了一个基于SM2的双方共同签名协议,增加了盲签名的特性. Yang等人[11]在2017年公开了一个基于SM2的双方共同签名协议,需要两轮通信和3次标量乘. Ding等人[2]在2018年提出了基于SM2的双方共同签名协议和一个门限群签名协议. 这两个协议先对私钥进行计算,然后再分享秘密份额,属于新的设计思路.

上述协议有些是专利技术,没有安全性分析,有些在论文中仅提供了非正式的安全分析,缺乏严格的安全性证明. He等人[12]给出了一个双方共同SM2签名协议. 该协议基于同态加密和零知识证明构造,给出了安全模型和安全性证明,且签名双方只需要一次通信,是目前具有安全性证明的较为高效的协议. 我们注意到其它典型签名算法的双方共同签名协议在提供了安全性证明之后,构造上也较为复杂. 例如Gilboa[13]给出了一个双方共同生成RSA模数和私钥的协议,使用了不经意传输和同态加密等算法. MacKenzie和Reiter[14]提出了一个可证安全的关于DSA数字签名算法的双方共同签名协议,在构造上采用了零知识证明和同态加密算法. Lindell[15]给出了一个可证安全的关于ECDSA的双方共同签名协议,在构造上使用了Paillier同态加密和零知识证明.

1.2　主要贡献

本文提出了一个新的SM2共同签名协议. 该协议首先是可证明安全的,然后该协议需要两轮通信和4次标量乘,没有同态加密或者零知识证明的计算需求,性能与不提供安全性证明的协议接近. 在使用预计算技术的情况下,服务器端在线完成一次两方共同签名,仅需1次标量乘运算.

2　背景知识

定义p为大素数,F_p为有限域. 选择$a,b\in F_p$作为椭圆曲线E的参数,定义P为椭圆曲线E上的一点,并且将其作为群G的生成元. 群G的阶为q. 定义$H_v()$为摘要长度为v比特的哈希算法.

2.1　SM2签名算法

SM2签名算法[6]包含了以下四个步骤.

(1) 初始化(Setup):给定安全参数λ,生成椭圆曲线参数$params=(p,a,b,P,q)$并输出.

(2) 密钥生成(Key):给定参数后,选择随机数$x\in Z_q^*$作为私钥,计算$Q=x\cdot P$作为公钥,输出(Q,x)作为公私钥对.

(3) 签名算法(Sign):给定参数$params$,私钥x以及消息m,签名者将执行以下步骤:

① 计算$Z=H_v(ENTL\|ID_A\|a\|b\|P\|pk)$,其中$ID_A$是用户的可辩别标识,$ENTL$是$ID_A$的长度,计算$\overline{M}=Z\|M$;

② 计算$e=H_v(\overline{M})$;

③ 选择随机数$k\in Z_q^*$,计算椭圆曲线点$R=k\cdot P=(x_1,y_1)$;

④ 计算$r=x_1+e\ mod\ q$,$s=(1+x)^{-1}(k-r\cdot x)\ mod\ q$;

⑤ 输出签名$\delta=(r,s)$.

(4) 验证算法(Ver):给定参数$params$,公钥$Q=x\cdot P$,消息m'以及它的签名$\delta'=(r',s')$,验证算法运行如下:

① 检验$r',s'\in Z_q^*$是否成立;

② 设$\overline{M}'=Z\|M'$,计算$e'=H_v(\overline{M}')$;

③ 计算$t=r'+s'$,若$t=0$则验证不通过;否则计算椭圆曲线点$R'=s'\cdot P+t\cdot Q=(x_1',y_1')$,验证$r'=e'+x_1'\ mod\ q$是否成立. 验证通过则输出1,否则输出0.

2.2　数字签名算法安全模型

数字签名算法的安全模型中包含挑战者C和敌手Adv. 挑战者与敌手运行如下的攻击游戏.

定义1.　数字签名算法攻击游戏给定一个数字签名算法(Setup, Key, Sign, Ver). 挑战者C运

行Setup和Key算法并把生成的系统参数para和验证公钥vk给敌手Adv. Adv,可以进行m次训练,即输入的消息为$(M_1,\cdots,M_m)$,由挑战者给出合格的数字签名$(\delta_1,\cdots,\delta_m)$. 在训练后,如果敌手Adv能够给出一个有效的消息签名对(M^*,δ^*),并且消息$M^*\notin\{M_1,\cdots,M_m\}$,那么我们就称敌手成功地伪造了签名,并将该事件称为$Event_{Adv}$. 可以用如下公式表示:

$$Event_{Adv}=\begin{Bmatrix}(\delta_1,\cdots,\delta_m)\leftarrow Adv^{C(\mathrm{sk},\cdot)}(M_1,\cdots,M_m)\\(M^*,\delta^*)\leftarrow Adv(M_1,\cdots,M_m,\delta_1,\cdots,\delta_m)\\M^*\notin\{M_1,\cdots,M_m\}\\1\leftarrow \mathrm{Ver}(M^*,\delta^*)\end{Bmatrix}$$

定义2. 在定义1所述的攻击游戏中,如果事件$Event_{Adv}$出现的概率可以忽略,那么数字签名算法具有不可伪造性.

2.3 知识提取假设

Damgård等人在1991年提出了一个假设[16],即知识提取假设版本1(KEAv1),该假设定义如下:

定义3. (KEAv1)设T是任意多项式时间算法,输入$(P,\alpha P)$,生成$(\beta P,\beta\alpha P)$,其中β是T选择的,那么存在另外一个多项式时间算法T^*,输入与T相同,使用的随机纸带与T相同,但是能够以$1-\sigma$的概率输出$(\beta,\beta P,\beta\alpha P)$,其中$\sigma$是可以忽略的量.

KEAv1在一般化的群表示模型中得到了验证[17],这为它的合理性提供了证据.

2.4 共同签名协议安全模型

协议中由两个实体Alice和Bob来完成签名. 为了更清晰地描述签名过程,我们将Alice运行的协议部分定义为Π_A,将Bob运行的协议部分定义为Π_B. 我们假设攻击者为Eve. Eve可以通过指令或者消息来触发Π_A或Π_B,或者攻击者可以腐化其中一方的诚实参与者,获得该参与者的所有信息,并模仿其与另一个诚实参与者签署协议. 我们将在随机预言机模型下证明其安全性,下面给出两个定义.

定义4. (共同签名协议攻击游戏)给定两个诚实参与者,Alice和Bob. 我们将一个基于SM2的双方共同签名协议$\prod$定义为$\left(\prod_A,\prod_B\right)$. 在Eve最多腐化一个诚实参与者b的情况下,$b\in\{A,B,\perp\}$. Eve可以进行m次训练,即输入的消息为$(M_1,\cdots,M_m)$,与诚实参与者执行签名协议交互,然后得到相应的数字签名$(\delta_1,\cdots,\delta_m)$. 在训练后,如果攻击者Eve能够给出一个消息签名对(M^*,δ^*),并且消息$M^*\notin\{M_1,\cdots,M_m\}$,且签名$\delta^*$的生成过程中未曾与诚实参与者的签名协议交互,那么我们就称攻击者成功地伪造了签名,并将该事件称为$Event_F$. 可以用如下公式表示:

$$Event_F=\begin{Bmatrix}\Pi=(\prod_A,\prod_B)\\b\in\{A,B,\perp\}\\(\delta_1,\cdots,\delta_m)\leftarrow Eve^{\{\Pi\}_{\{A,B,\perp\}\backslash\{b\}}}(M_1,\cdots,M_m)\\(M^*,\delta^*)\leftarrow Eve(M_1,\cdots,M_m,\delta_1,\cdots,\delta_m)\\M^*\notin\{M_1,\cdots,M_m\}\\(M^*,\delta^*)\text{有效}\end{Bmatrix}$$

在定义4中,攻击者可能会从Alice和Bob中选择一个进行腐化,也可能不腐化任何诚实参与者,当攻击者腐化其中某一个诚实参与者之后,攻击者就只需跟另外一个参与者交互即可,当攻击者没有腐化诚实参与者时,攻击者就要跟两个诚实参与者交互,表示为$\{\Pi\}_{\{A,B,\perp\}\backslash\{b\}}$. 当攻击者腐化Alice参与者时,攻击者需要通过向Bob发送第一轮消息,通过消息触发Bob进入一轮协议的执行,并尝试伪造签名. 当攻击者腐化Bob参与者时,攻击者通过程序调用的方式向Alice发送待签名消息,触发Alice进入一轮协议的执行,并尝试伪造签名.

消息$(M_1,\cdots,M_m)$由攻击者Eve自适应选择,即消息M_i的选择取决于前$i-1$次交互的情况.

接下来,我们对不可伪造性给出定义.

定义5. 在定义4所述的攻击游戏中,如果事件$Event_F$出现的概率可以忽略,那么签名协议$\prod$具有不可伪造性.

3 共同签名协议

共同签名协议包括Alice和Bob两方,我们设定签名是由Alice发起的. 具体协议可以分为密钥生成和签名两个子协议. 以下对协议进行详细描述.

(1)共同密钥生成

① Alice随机秘密选择$d_A\in Z_q^*$,计算$pk_A=d_A\cdot P$,然后Alice发送pk_A给Bob.

② Bob随机秘密选择$d_B\in Z_q^*$,计算$pk_B=d_B\cdot P$,然后Bob发送pk_B给Alice. 计算$pk=d_Ad_BP-P$,pk是Alice和Bob的共同公钥,d_A和d_B分别是Alice和Bob的部分私钥,pk_A和pk_B分别是Alice和Bob的公钥分量. Alice和Bob分别存储(pk,d_A,pk_B)和(pk,d_B,pk_A).

2)共同签名生成

① Alice选择一个随机数$k_A \in Z_q^*$,并计算$R_A = k_A \cdot P$和$R_A' = k_A \cdot pk_B$,向Bob发送R_A、R_A'.

② Bob收到消息后,验证$R_A = d_B \cdot R_A'$,验证失败则退出协议,否则选择一个随机数$k_B \in Z_q^*$,计算$R_B = k_B \cdot pk_A$,$R_B' = k_B \cdot P$,并发送R_B、R_B'给Alice.

③ Alice收到消息后,验证$R_B = d_A \cdot R_B'$,验证失败则退出协议,否则计算椭圆曲线群元素$R' = R_A + R_B = (x_A, y_A)$,计算$r = H(Z_A \| M) + x_A \bmod q$和$s' = (k_A + r) d_A^{-1} \bmod q$,并将s′发送给Bob.

④ Bob收到s′以后计算$t = (s' + k_B) d_B^{-1} \bmod q$,并将$t$发送给Alice.

⑤ Alice收到t以后,计算$s = t - r$,输出数字签名(r, s).

3.1　正确性分析

根据SM2数字签名的验证算法,只需要计算椭圆曲线点$R' = s' \cdot P + t \cdot Q = (x_1', y_1')$,并验证$r' = e' + x_1' \bmod q$是否成立. 表1中给出了协议的直观描述,可以看到,我们的协议具有以下恒等关系:

$$\begin{aligned} R' &= s \cdot P + t \cdot pk \\ &= (t - r)P + t(d_A d_B P - P) \\ &= tp - rp + td_A d_B P - tp \\ &= -rp + ((s' + k_B) d_B{}^{-1}) d_A d_B P \\ &= -rp + (s' + k_B) d_A P \\ &= -rp + ((k_A + r) d_A{}^{-1} + k_B) d_A P \\ &= -rp + (k_A + r)P + k_B d_A P \\ &= k_A P + k_B P K_A \\ &= R_A + R_B \end{aligned}$$

因此,本协议的正确性可以得到验证.

3.2　效率分析

因为目前具有安全性证明的SM2共同签名协议只有He等人[12]的提议,表2中给出三个协议的效率分析. 定义Hom表示一次Parillier加密或者解密运算,ZK表示一次零知识证明或者验证运算,Exp表示一次模幂运算或一次标量乘运算. 当某一方所做运算与对方的消息或者待签名的消息无关时,该运算可以预先进行,通常这部分运算称为线下运算. 与之对应的,则是线上运算. 通常认为线上运算是一个更为关键的计算考量指标. 因此我们单独列出了线上运算所需的运算量.

表1　共同签名生成协议

Alice		*Bob*
计算$R_A = k_A \cdot P$, $R_A' = k_A \cdot pk_B$		
	$\xrightarrow{R_A, R_A'}$	验证$R_A = d_B \cdot R_A'$ 计算$R_B = k_B \cdot pk_A$ $R_B' = k_B \cdot P$
	$\xleftarrow{R_B, R_B'}$	
验证$R_B = d_A \cdot R_B'$ 计算$R' = R_A + R_B = (x_A, y_A)$ 计算 $Z_A = H(ENTL \| ID_A \| a \| b \| P \| pk)$ 计算$r = H(Z_A \| M) + x_A \bmod q$ 计算$s' = (k_A + r) d_A^{-1} \bmod q$		
	$\xrightarrow{s'}$	
		计算$t = (s' + k_B) d_B^{-1} \bmod q$
	$\xleftarrow{t}$	
计算$s = t - r$ 输出签名(r, s)		

在没有安全性证明的共同签名协议中,效率最高的协议在林等人[8]的专利中. 他们的协议只需要一轮通信,Alice需要1次Exp,Bob需要2次Exp. 考虑预计算后,Alice不需要线上的Exp运算,Bob只需要1次Exp运算. 本文的协议在效率上与林等人[8]的协议接近.

表2　效率对比

	消息轮数	主要计算量		线上主要计算量	
		Alice	Bob	Alice	Bob
He等人[12]	1	2Exp+ 2Hom+ 2ZK	4Exp+ 2Hom+ 2ZK	1Exp+ 1Hom+ 1ZK	3Exp+ 2Hom+ 1ZK
本文	2	2 Exp	2 Exp	1 Exp	1 Exp
Lin等人[8]	1	1Exp	2Exp	0	1Exp

4　安全性分析

在这一节中,我们将对共同签名协议的安全性进行一个正式的分析. 在第二章中,我们已经给出了安全模型.

4.1 安全性证明

定理1. 设KEAv1成立的概率是$1-\varepsilon_{KEAv1}$，攻击者可以获得m条消息的共同签名，询问随机预言机的次数最多为o次，随机预言机的输出空间为n，攻击者攻击共同签名协议成功的概率为ε_{Eve}，那么存在仿真算法*SM2.Sim*能够成功伪造SM2数字签名的概率至少为$\frac{2}{3}(1-\frac{o}{n})(1-\frac{2}{3}m\varepsilon_{KEAv1})\varepsilon_{Eve}$.

证明. SM2数字签名体制包含(SM2. Setup, SM2. Key, SM2. Sign, SM2. Ver)等算法，并且定义一个SM2体制的攻击者*SM2.Sim*. 该攻击者仿真签名协议$\prod$中的两个诚实参与者，与攻击者Eve进行交互，希望Eve在生成伪造的签名时，能够同步地生成针对原SM2体制的数字签名. *SM2.Sim*从*SM2.Setup*和*SM2.Key*算法中获取系统参数和验证密钥vk. 之后*SM2.Sim*将vk作为共同公钥，并且随机选择$sim.b\in\{A,B\}$，选择私钥$sk_{sim.b}\in Z_q^*$. 如果$sim.b=A$，*SM2.Sim*就设置$d_A=sk_{sim.b}$，否则设置$d_B=sk_{sim.b}$. 当$sim.b=A$时，此时$d_B=\log_P(vk+P)^*\left(sk_{sim.b}\right)^{-1}$. 由此可知，*SM2.Sim*显然只能模拟Alice和Bob其中一方的密钥，而不知道另一方的密钥.

下面描述*SM2.Sim*和Eve的交互过程. 我们假设Eve的交互是按签名步骤进行的：

(1)首先Eve选择$b\in\{A,B,\perp\}$，这里可分为三种情况，如果$b=sim.b$，那么*SM2.Sim*就把自己所仿真的诚实参与者的状态和私钥交给Eve，并且停止对参与者$sim.b$的仿真；如果$b=\perp$，则*SM2.Sim*自己仿真两个诚实参与者，与Eve进行交互；若$b\neq sim.b$且$b\neq\perp$，则*SM2.Sim*仿真游戏失败.

(2)在第一种情况中，又可分为两种情况进行讨论

1) 如果$b=sim.b=B$，那么*SM2.Sim*随机选择私钥分量$d_B\in Z_q^*$，并将(d_B,pk,pk_A)提供给Eve. 而*SM2.Sim*需要向Eve仿真Alice的协议，*SM2.Sim*拥有(pk,pk_B)，其中$pk_B=d_B P$.

① *SM2.Sim*维护一个随机预言机，该预言机输入为有限长的字符串，当输入为$Z_A\|M_i$时，则返回保存的对应哈希值，如果没有记录则停止游戏. 注意到本文的共同签名协议中只有Alice一方需要计算哈希函数.

② 当Eve要求Alice对消息M_i进行签署时，*SM2.Sim*首先向SM2挑战者提交M_i，然后获得消息M_i的数字签名(r_i,s_i). 之后*SM2.Sim*仿真Alice形成第一条消息，即选择一个随机数$k_A\in Z_q^*$，并计算$R_A=k_A\cdot P$和$R_A'=k_A\cdot pk_B$，向Eve发送R_A、R_A'.

③ 当*SM2.Sim*收到Eve返回的第一条信息(R_B,R_B')后，*SM2.Sim*把Eve看作算法T，输入为(P,pk_A)，输出为(k_BP,k_Bpk_A)，根据KEAv1，此时存在一个算法T^*，在输入(P,pk_A)后，能够输出(k_B,k_BP,k_Bpk_A)，即可以获得k_B. 如果该过程运行失败，就说明Eve没有诚实生成该消息，则*SM2.Sim*停止协议执行. 若运行成功，则*SM2.Sim*计算r和s'，由于*SM2.Sim*没有Alice的私钥分量d_A，所以通过挑战者返回的数字签名来计算s'，即$s'=(r_i+s_i)d_B-k_B$，并将s'返回给Eve. 同时*SM2.Sim*设置随机预言机对$Z_A\|M_i$的响应为r_i-x_A.

④ 当*SM2.Sim*收到Eve返回的第二条消息t时，*SM2.Sim*检验t是否等于s_i+r_i，如果不相等则退出协议，否则输出(r_i,s_i)作为对消息M_i的签名.

2) 如果$b=sim.b=A$，那么*SM2.Sim*随机选择私钥分量$d_A\in Z_q^*$，并且将(d_A,pk,pk_B)提交给Eve. 而*SM2.Sim*需要向Eve仿真Bob的协议，*SM2.Sim*拥有(pk,pk_A)，其中$pk_A=d_A P$.

① *SM2. Sim*维护一个随机预言机，该预言机输入为有限长的字符串，当输入为$Z_A\|M_j$时，则返回保存的对应哈希值，如果没有记录则向SM2挑战者提交消息M_j并且请求签名，然后获得消息M_j关于私钥vk的数字签名(r_j,s_j)，如果此时*SM2. Sim*已经与攻击者完成了该次会话的第一轮消息交互，则计算$(x_A,y_A)=R_A+R_B$，并设置随机预言机对$Z_A\|M_j$响应为r_j-x_A. 否则停止游戏. 注意到按照协议规范，Alice一方计算哈希函数的时机是在第一轮消息交互完成之后.

② 当*SM2.Sim*收到Eve发送第一条消息R_A、R_A'时，假设Eve诚实地执行了协议，那么此时*SM2.Sim*把Eve看作算法T，输入为(P,pk_B)，输出为(k_AP,k_Apk_B)，根据KEAv1，此时存在一个算法T^*，在输入(P,PK_A)后，能够输出(k_A,k_AP,k_Apk_B)，即可以获得k_A. 如果该过程运行失败，就说明Eve没有诚实生成该消息，则*SM2.Sim*停止协议执行. 若运行成功，则*SM2.Sim*选取随机数$k_B\in Z_q^*$，计算R_B和R_B'，并发送(R_B,R_B')给Eve.

③ 当*SM2.Sim*收到Eve的第二条消息s′后，需要给Eve返回响应t. 但注意到*SM2.Sim*并没有Bob

的私钥分量来计算t,因此*SM2.Sim*找到之前SM2挑战者为本轮消息M_j生成的SM2数字签名(r_j, s_j),并发送$r_j + s_j$给Eve作为响应.

3) 如果$b=\perp$,那么*SM2.Sim*需要仿真Alice或者Bob的协议. 若$sim.b=A$,*SM2.Sim*随机选择$d_B \in Z_q^*$,并诚实地运行Bob部分的协议,此时交互过程大致与步骤2)类似,此时不再需要通过KEAv1来提取Bob的随机数k_B;若$sim.b=B$,*SM2.Sim*随机选择$d_A \in Z_q^*$,并诚实地运行Alice部分的协议,此时交互过程大致与步骤1)类似,此时不再需要通过KEAv1来提取Alice的随机数k_A.

下面我们对上述签名过程进行分析.

当Eve经过充分的训练,给出一个伪造的签名对(M^*, δ^*)之后,由于公钥和系统参数都相同,*SM2.Sim*就能够把该签名作为SM2算法的有效伪造签名. 下面我们分析*SM2.Sim*成功的概率.

首先我们需要确认Eve已经受到合格的训练,难以区分*SM2.Sim*提供的仿真环境和真实的与Alice和Bob交互的环境. 当然即使是攻击者与真实的Alice和Bob交互,我们也可以要求攻击者、Alice、Bob的所有哈希操作都需要经过随机预言机的查询才能实现. 攻击者最多询问m个消息的签名,最多o次查询随机预言机,设哈希函数的输出空间为n,那么攻击者未经随机预言机查询而得到一个合格的输出的概率是$1/n$,那么在整个攻击过程中,攻击者最多有o/n的概率能区分随机预言机的输出,那么不能区分的概率就是$1-o/n$.

在Eve与*SM2.Sim*交互时,*SM2.Sim*会在Eve询问随机预言机的时机不对时会停止游戏. 这是由协议实际运行时,哈希函数计算的时机和执行计算的实体决定的,我们在概率分析时不作为仿真者失败的概率.

在Eve与*SM2.Sim*交互时,*SM2.Sim*在按照KEAv1假设提取随机数失败时会停止游戏,这与协议实际运行时,Alice或者Bob会检查对方是否有按照协议规范运行的过程是一致的,同样不作为仿真者失败的概率.

下面,我们断言攻击者的所有哈希操作都需要经过随机预言机查询得到,然后继续分析*SM2.Sim*成功的概率. *SM2.Sim*事先并不清楚Eve腐化的对象是Alice、Bob还是不腐化,因此*SM2.Sim*有仿真游戏失败的可能. 我们用$Event_0$来表示*SM2.Sim*仿真失败的事件,$Event_1$表示*SM2.Sim*选择$sim.b=A$的事件,$Event_2$表示*SM2.Sim*选择$sim.b=b$的事件,$Event_3$表示Eve腐化Alice的事件,$Event_4$表示Eve腐化Bob的事件,假设这四个事件发生的概率服从均匀分布,那失败的概率为:

$$\begin{aligned} Pr(Event_0) &= Pr(Event_1 \wedge Event_4) + \\ &\quad Pr(Event_2 \wedge Event_3) \\ &= \frac{1}{2} \times \frac{1}{3} + \frac{1}{2} \times \frac{1}{3} \\ &= \frac{1}{3} \end{aligned}$$

那么这一步成功的概率就是2/3.

下面我们继续假设*SM2.Sim*在第一步仿真游戏是成功的,然后继续分析*SM2.Sim*成功的概率. 我们用$Event_5$、$Event_6$、$Event_7$来分别表示步骤1)、2)、3)来进行分析. 显然,任何一个事件中若Eve能够给出伪造签名都会导致*SM2.Sim*的伪造成功,因此伪造成功的概率可以描述为:

$$Pr(Event_F) = Pr(Event_5) + Pr(Event_6) + Pr(Event_7)$$

对于$Event_5$,Eve的每一次签名请求,在整个仿真过程中有三个地方与实际执行过程不同:(1)步骤1)中的①子步骤使用了随机预言机,对于引入随机预言机造成的差异,我们已经在分析开始的部分进行了统一的分析,因此差异(1)可以在此忽略;(2)步骤1)中的③子步骤采用KEAv1得到的k_B,k_B由KEAv1得到,如果KEAv1不成立的概率是ε_{KEAv1},那么差异(2)不可区分的概率为$1-\varepsilon_{KEAv1}$;(3)步骤1)中的③子步骤没有用Alice的私钥分量生成s',而是通过SM2挑战者返回的数字签名和k_B来计算s',*SM2.Sim*生成s′的概率分布取决于SM2挑战者的响应,在SM2签名算法中,若密钥确定,则其随机性取决于一个$\{1, \cdots, q-1\}$中的随机数;在实际执行过程中,Alice计算s′的过程中,随机性取决于k_A,即一个$\{1, \cdots, q-1\}$中的随机数,因此就随机性上并没有引入新的概率差异.

当Eve不能区分*SM2.Sim*的仿真签名和真实签名过程时,在询问m次后,按照假设最终Eve给出一个伪造的签名(M^*, δ^*),且$M^* \notin \{M_1, \cdots, M_m\}$,共同公钥为$vk$. 此时$(M^*, \delta^*)$是一个有效的SM2伪造签名.

设攻击者在充分训练后成功的概率是ε_{Eve},那么

$$\begin{aligned} Pr(Event_5) &\geqslant \frac{2}{3} \cdot \frac{1}{3} \cdot (1 - m \cdot \varepsilon_{KEAv1}) \varepsilon_{Eve} \\ &= \frac{2}{9} (1 - m\varepsilon_{KEAv1}) \varepsilon_{Eve} \end{aligned}$$

对于Event_6，Eve的每一次签名请求，与Event_5情况类似，那么

$$Pr(Event_6)\geqslant\frac{2}{3}\cdot\frac{1}{3}\cdot(1-m\cdot\varepsilon_{KEAv1})\varepsilon_{Eve}$$
$$=\frac{2}{9}(1-m\varepsilon_{KEAv1})\varepsilon_{Eve}$$

对于$Event_7$，Eve的每一次签名请求，与$Event_5$情况类似，但不需要通过KEAv1来得到k_B，与真实情况相比没有第二项差异，因此

$$Pr(Event_7)\geqslant\frac{2}{3}\cdot\frac{1}{3}\cdot\varepsilon_{Eve}=\frac{2}{9}\varepsilon_{Eve}$$

综合考虑随机预言机的影响，并且KEAv1不成立的概率ε_{KEAv1}是可以忽略不计的量，则可以得到：

$$Pr(Event_F)=Pr(Event_5)+Pr(Event_6)+Pr(Event_7)$$
$$\geqslant(1-\frac{o}{n})(\frac{2}{9}(1-m\varepsilon_{KEAv1})\varepsilon_{Eve}$$
$$+\frac{2}{9}(1-m\varepsilon_{KEAv1})\varepsilon_{Eve}+\frac{2}{9}\varepsilon_{Eve})$$
$$\geqslant(1-\frac{o}{n})(\frac{2}{3}\varepsilon_{Eve}-\frac{4}{9}m\varepsilon_{KEAv1}\varepsilon_{Eve})$$
$$=\frac{2}{3}(1-\frac{o}{n})(1-\frac{2}{3}m\varepsilon_{KEAv1})\varepsilon_{Eve}$$

根据定理可以知道，如果存在攻击者能够伪造共同签名，那么这个攻击者也就能够伪造SM2数字签名；也就是说，如果SM2数字签名算法是安全的，那么我们的共同签名协议也是安全的.

5 应用协议

共同签名协议在实际运行时会出现新的需求，例如需要为共同公钥颁发数字证书，以及Bob一方需要认证Alice一方后才能使用Bob存储的私钥分量. 下面给出一个应用协议，能够给出共同公钥证书的生成过程和具有验证Alice一方功能的共同签名协议，使得共同签名协议可以切实地应用在实践中.

5.1 协议描述

该协议分为两个部分，首先是共同签名公钥证书的生成子协议，之后是具有对第二轮消息验证过程的数字签名生成子协议.

(1)共同签名公钥证书

该过程包括CA，Alice和Bob三方. Alice一方生成证书签名请求(CSR)，获取数字证书.

1) Alice随机秘密选择私钥分量$d_A\in Z_q^*$，计算Alice的公钥分量$pk_A=d_A\cdot P$，然后Alice发送pk_A给Bob.

2) Bob随机秘密选择私钥分量$d_B\in Z_q^*$，计算Bob的公钥分量$pk_B=d_B\cdot P$，选择一个随机数$k_B^0\in\{1,\cdots,q-1\}$，计算第一个随机群元素$R_B^0=k_B^0\cdot pk_A$，然后Bob发送R_B^0和pk_B给Alice.

3) Alice选择随机数$k_A^0\in\{1,\cdots,q-1\}$，计算$R_A^0=k_A^0+R_B^0$，设$R_A^0=(x_A^0,y_A^0)$，Alice计算共同公钥$pk=d_A pk_B-P$，提供身份信息以在客户端生成证书签名请求CSR的内容，设为M_{CSR}，计算$r=H(Z_A\|M_{CSR})+x_A^0 \bmod n$，计算$s'=(k_A^0+r)d_A{}^{-1}$，然后把$M_{CSR}$和$s'$提交给Bob.

4) Bob验证M_{CSR}的内容，核对身份，失败则退出协议，否则计算$t=(s'+k_B^0)d_B{}^{-1}$，并向Alice返回t.

5) Alice计算签名$(r,t-r)$，合成完整的CSR，向Bob提交.

6) Bob验证完整的CSR信息是否与d)步骤中的M_{CSR}的内容相符合，验证不通过则退出，否则向CA提交.

7) CA验证CSR，颁发共同公钥证书，存储到证书库中.

8) Alice或者Bob都可以获取证书库中的共同公钥证书，Alice在本地存储该共同公钥证书和Bob的公钥分量pk_B，Bob在本地存储该共同公钥证书和Alice的公钥分量pk_A.

(2)验证加强的共同签名协议

该过程中Bob需要先对Alice进行验证才能使用Bob存储的部分私钥，因此能够更好地满足Bob的信息安全策略. 协议依旧包括Alice和Bob两方，由Alice发起.

1) Alice选择一个随机数$k_A\in\{1,\cdots,q-1\}$，并计算$R_A=k_A\cdot P$和$R_A'=k_A\cdot pk_B$，其中pk_B来自本地存储，计算完成后发送pk_A的哈希值，R_A和R_A'给Bob.

2) Bob使用私钥分量d_B验证$R_A=d_B\cdot R_A'$，验证失败则退出协议，否则选择一个随机数$k_B\in\{1,\cdots,q-1\}$，计算$R_B=k_B\cdot pk_A$，$R_B'=k_B\cdot P$，并发送R_B、R_B'给Alice. 其中pk_A由pk_A的哈希值索引.

3) Alice验证$R_B=d_A\cdot R_B'$，如果验证失败则退出协议，否则计算椭圆曲线群元素$R'=R_A+R_B$，由于$R'=(x_A,y_A)$，可以计算$\mathrm{r}=\mathrm{H}(Z_A\|M)+x_A \bmod q$和$\mathrm{s}'=(k_A+r)d_A^{-1} \bmod q$，之后使用私钥

分量计算关于$(R_A, R_A', R_B, R_B', s')$的普通SM2数字签名$\delta_A$,并把s′和$\delta_A$发送给Bob.

4)Bob重构签名消息,使用Alice的公钥分量pk_A验证Alice的数字签名δ_A,验证不通过则退出,否则使用随机数k_B和收到的s′计算$t=(s'+k_B)d_B^{-1} \bmod q$,并发送$t$给Alice.

5)Alice合成SM2数字签名$(r, t-r)$,验证数字签名的正确性,验证无效则退出协议,否则获得签名.

5.2　协议分析

相较于共同签名协议,应用协议是对效率和安全性一个折中.协议首先在共同公钥生成阶段中给出了和CA证书颁发流程一致的协议,并且明确要求了Alice向Bob认证身份,可以有效防止Alice提供虚假信息;在共同签名时,实现了对签名者的身份认证,能够符合Bob一方的安全策略.同时计算上签名双方也只需分别进行两次在线的标量乘运算,计算代价适中.

与表1所呈现的共同签名协议相比,共同签名公钥证书子协议在第一轮消息交换中减少了一个随机群元素的计算和验证,在第二轮消息交换中增加了证书签名请求内容的交换,并在第三轮收到客户端证书请求后进行交叉验证.首先从定理1的证明过程我们可以看到Alice一方生成的随机群元素在证明过程中仅仅起到了验证的作用,并没有参与随机预言机的响应中,因而缺少该元素只会使得证明中2)的②步骤缺少验证,但并不会使得证明出现困难.因此,该元素的确实并不会导致安全问题.事实上,Bob一方通过对证书请求内容的核验可以终止协议,形式上是对表1中Bob能力的一种补偿.基于这些分析,我们给出下述结论.

推论1.　如果表1所呈现的共同签名协议是是不可伪造的,那么共同签名公钥证书子协议可以看作是应用上述协议产生证书请求的过程,也具有不可伪造性.

与表1所呈现的共同签名协议相比,验证加强的共同签名子协议保留了共同签名协议的全部消息,并增加了一次Alice使用私钥分量生成数字签名的过程,这样处理纯粹是为了符合Bob的安全策略,对协议本身的安全性没有影响.我们给出以下推论:

推论2.　如果表1所呈现的共同签名协议是是不可伪造的,那么验证加强的共同签名子协议保留了原有的所有消息,并增加了一次数字签名生成和验证的工作,也是不可伪造的.

6　结　论

本文首先给出了一个新的基于SM2的共同签名协议,并且给出了安全性证明.与当前可证安全的类似协议相比,我们的协议极大的缩小了计算代价,通信代价则需要额外的1轮通信.考虑到实际中数字证书颁发和服务器安全策略的问题,我们给出了一个应用协议.

致　谢　衷心感谢审稿专家和编辑部老师提出的宝贵意见和建议!

参　考　文　献

[1] The 43rd CNNIC China Internet Report. China Broadcasts, 2019, 04: 48 (in Chinese)
(第43次CNNIC中国互联网报告发布.中国广播, 2019, 04: 48)

[2] Ding F, Long Y H, Wu P. Study on Secret Sharing for SM2 Digital Signature and Its Application//Proceedings of the 2018 14th International Conference on Computational Intelligence and Security (CIS). Hangzhou, China, 2018: 205-209

[3] Blömer J. How to Share a Secret. Communications of the ACM, 2011, 22(22): 612-613

[4] Lai Xi-Song, Han Liang, Zhang Zhen-Cheng. Computer cryptography and applications. Beijing: National Defence Industrial Press, 2001
(赖溪松, 韩亮, 张真诚. 计算机密码学及其应用. 北京: 国防工业出版社, 2001)

[5] Chen Li-Quan, Zhu Zheng, Wang Mu-Yang, et al. A Threshold Group Signature Scheme for Mobile Internet Application, Chinese Journal of Computers, 2018, 425(05): 86-101 (in Chinese)
(陈立全, 朱政, 王慕阳等. 适用于移动互联网的门限群签名方案. 计算机学报, 2018, 425(05): 86-101)

[6] Chinese Encryption Administration. GM/T 0003-2012 SM2 Elliptic Curve Public-Key Cryptography Algorithm. Beijing, China, 2010

[7] Halderman J A, Heninger N, Moore J, et al. Elliptic Curve Cryptography in Practice//Proceedings of the 18th International Conference on Financial Cryptography and Data Security. Christ Church, Barbados, 2014: 157-175

[8] Lin Jin-Qiang, Ma Yuan, Jing Ji-Wu, et al. Signing and decryption methods and systems based on SM2 scheme suitable for cloud computation, China, 2014.12.24 (in Chinese)
(林璟锵, 马原, 荆继武等. 适用于云计算的基于SM2算法

的签名及解密方法和系统，中国专利，2014.12.24）

[9] Jie Y，Yu L，Li-Yun C，et al. A SM2 elliptic curve threshold signature scheme without a trusted center. KSII Transactions on Internet and Information Systems，2016，10(02)：897-913.

[10] Zhang Yong-Qiang，Liu Qiang，Collaberative singing and decryption methods，devices and systems of SM2 scheme，China，2017.09.22（in Chinese）
（张永强，刘镪，SM2协同签名及解密方法、装置与系统，中国专利，2017.09.22）

[11] Yang Guo-Qiang，Liu Hui-Yi，A distributed signature method and system based on elliptic curve，China，2017.05.17（in Chinese）
（杨国强、刘会议，一种基于椭圆曲线的分布式签名方法及系统，中国专利，2017.05.17）

[12] Zhang Y. D.，He D. B.，Zhang M. W.，et al. A Provable-Secure and Practical Two-Party Distributed Signing Protocol for SM2 Signature Algorithm，Frontiers of Computer Science，http://journal.hep.com.cn

[13] Gilboa N . Two Party RSA Key Generation// Proceedings of the Advances in Cryptology. Berlin，Germany，1999：116-129

[14] Mackenzie P，Reiter M K. Two-Party Generation of DSA Signatures//Proceedings of the Annual International Cryptology Conference. California，USA，2001：137-154

[15] Lindell Y. Fast Secure Two-Party ECDSA Signing// Proceedings of the Annual International Cryptology Conference. California，USA，2017：613-644

[16] Damgård I. Towards practical public key systems secure against chosen ciphertext attacks// Proceedings of the Annual International Cryptology Conference. California，USA，1991：445-456

[17] Dent A W，Galbraith S D. Hidden Pairings and Trapdoor DDH Groups Algorithmic Number Theory//Proceedings of the International Algorithmic Number Theory Symposium. Berlin，Germany，2006：436-451

Su Yin-Xue，M. S. candidate. Her current research interests include information security，cryptographic protocol.

Tian Hai-Bo，Ph. D. associate professor. His main interests include design and analysis of security protocol，block chain technology.

Background

The problem studied in this paper is the security of the mobile terminal private key storage in cyberspace security. In view of this problem，there are two sets of ideas in the world，which are based on hardware and software. Among the software-based methods，two-party signing algorithms based on RSA，DSA，ECDSA and other classical algorithms have been presented internationally，but for SM2 algorithm，there is still a lack of safe and effective schemes. This paper presents the first safe and effective scheme which is close to the practical application and improves the computational efficiency greatly when compared with the previous provable secure scheme. The main project of this subject is the research on the new principle and new algorithm of electronic currency in the 2017 national key research and development project，which mainly focuses on the technical issues of legal tender issued by the country. The result of this paper belongs to the security of the client wallet in the process of electronic money circulation.