基于SM2异步远程密钥生成的工业互联网账户恢复协议

肖浩[1,2,3]，杨雪[1]，姜奇[1,2]，余增文[2,4]，李兴华[1]，马建峰[1]
（1.西安电子科技大学网络与信息安全学院，陕西 西安 710071；2.大数据与决策实验室，湖南 长沙 410073；
3.海南核电有限公司，海南 昌江 572732；4.北京计算机技术及应用研究所，北京 100039）

摘　要：工业互联网相对开放的网络环境可能导致身份伪造和数据泄露等安全隐患，因而实现有效身份认证并确保用户账户安全至关重要。而现有工业互联网认证协议大多专注于认证阶段，对认证设备丢失后的账户恢复问题则缺乏重视。此外，国家对密码应用的自主可控也有明确要求。为此，提出了基于SM2异步远程密钥生成（ARKG）的账户恢复协议。首先，采用SM2盲化密钥封装机制和模糊提取器设计ARKG协议，实现了私钥与用户生物特征的绑定，并增强了派生私钥的安全性。基于该ARKG构造，提出工业互联网账户恢复协议，实现用户与服务器在协议恢复阶段的双向认证，并有效应对备份验证设备丢失问题。可证明安全分析与性能评估实验表明，所提协议可满足工业互联网的高安全性和可用性需求。

关键词：工业互联网；异步远程密钥生成；账户恢复；SM2
中图分类号：TN92
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2025092

SM2 asynchronous remote key generation based account recovery protocol in industrial Internet

XIAO Hao[1,2,3], YANG Xue[1], JIANG Qi[1,2], YU Zengwen[2,4], LI Xinghua[1], MA Jianfeng[1]
1. School of Cyber Engineering, Xidian University, Xi’an 710071, China
2. Laboratory for Big Data and Decision, Changsha 410073, China
3. Hainan Nuclear Power Co., Ltd., Changjiang 572732, China
4. Beijing Institute of Computer Technology and Application, Beijing 100039, China

Abstract: The openness network environment of the industrial Internet may lead to security risks such as identity forgery and data leakage, making it crucial to achieve effective authentication and ensure account security. However, existing industrial Internet authentication protocols mostly focus on the authentication phase and often overlooking the issue of account recovery in cases of lost authenticators. Moreover, there are explicit national requirements for autonomy and control in cryptographic applications. Therefore, an account recovery protocol based on SM2 asynchronous remote key generation (ARKG) was proposed. Firstly, the SM2 blind key encapsulation mechanism and fuzzy extractor were used to design the ARKG algorithms, which bound the private key to the user’s biometric features and enhances the security of derived private keys. Based on this ARKG construction, an industrial Internet account recovery protocol was proposed, achieving mutual authentication between the user and the server in the recovery phase and effectively addressing the issue of lost backup authenticators. Security analysis and experimental results demonstrate that the proposed protocol meets the high security and usability requirements of the industrial Internet.

Keywords: industrial Internet, ARKG, account recovery, SM2

收稿日期：2025–01–17；修回日期：2025–05–12
通信作者：姜奇，jiangqixdu@gmail.com
基金项目：国家自然科学基金资助项目（No.62472337, No.62072352, No.62125205, No.62372350）；陕西省杰出青年科学基金资助项目（No.2025JC-JCQN-084）
Foundation Items：The National Natural Science Foundation of China (No.62472337, No.62072352, No.62125205, No.62372350), The Natural Science Basic Research Program of Shaanxi Province (No.2025JC-JCQN-084)

0 引言

工业互联网是将物联网、大数据和人工智能等技术深度融合工业控制系统（ICS, industrial control system）所形成的网络体系。其核心目标是通过设备与系统互联，实现智能化生产、资源优化，显著提高生产灵活性，并推动工业升级。我国工业互联网规模持续扩大，虽带来智能化变革机遇，却也伴随更高的网络攻击风险，工业互联网始终是网络攻击者的重点目标。

工业互联网环境相对开放，信息交互依赖于公共传输通道，通信双方的可信度难以保障，易导致身份伪造和数据泄露等安全隐患[1-2]。攻击者可能通过中间人攻击或弱认证手段伪造合法身份，获取对系统的非法访问权限，从而泄露敏感信息泄露剥夺系统控制权。因此可靠的身份认证技术是确保系统安全的关键，可以有效防止未经授权的访问，确保账户安全。

国内外已对身份认证协议[3-4]展开大量研究，考虑到工业互联网系统实时性、跨平台互操作性、轻量化等方面的需求，现有方案[5-7]大多选择采用Web认证机制。例如，开放授权（OAuth, open authorization）[8]、开放身份（OpenID, open identification）、线上快速身份验证（FIDO, fast identity online）和FIDO2[9]等主流协议，通过验证器设备、生物特征等不同方式来验证用户身份。以上协议在工业互联网场景中也存在一定局限性，例如，OAuth和OpenID协议适用于多方授权场景，但依赖外部授权服务器，流程复杂；FIDO类协议具备较高的安全性与用户体验，适用于防止口令泄露和钓鱼攻击，但对终端设备的计算能力和硬件支持有较高要求，在设备种类多样、资源受限的工业场景下的适应性仍有待提升。

目前，针对工业互联网的身份认证机制的研究主要聚焦于认证阶段安全性，已相对较完善。但认证设备丢失及丢失后的账户恢复问题没有受到广泛关注与重视。账户丢失可能导致重要信息泄露、系统被滥用或业务中断，从而对企业造成严重损失。因此，账户恢复机制必不可少。该机制不仅需要确保用户身份的认证和恢复过程的安全性，还要考虑到用户体验，以便迅速有效地恢复账户，最大限度减少对业务的影响。

为防止用户用于认证的硬件设备丢失而无法访问账户，Yubico公司[10]提出允许用户连接2个硬件令牌，其中主验证设备可以代表备份验证设备生成公钥并提前向服务器进行注册。如果主验证设备丢失，用户可以通过备份验证设备恢复对账户的访问权。Frymann等[11]将该过程抽象为一个密码学原语，称为异步远程密钥生成（ARKG, asynchronous remote key generation），并基于DH（Diffie-Hellman）交换与密钥封装机制（KEM, key encapsulation mechanism）实例化。为兼容各类密码系统，研究人员提出基于双线性配对和格的ARKG构造[12-15]，在不同应用中对ARKG的使用提供参考。

尽管现有基于ARKG的账户恢复缓解了设备丢失风险，但仍存以下不足。首先，工业互联网作为商用密码应用的重点领域，应使用国密标准算法以确保信息的安全合规，而现有ARKG未采用国密标准算法设计，无法满足我国对网络空间安全自主化要求。其次，现有基于ARKG的账户恢复协议并未考虑到用户对服务器端的认证，降低了ICS的认证安全性。最后，采用备份验证设备提供防丢失保障的同时，也增大了敌手攻击面积，任何拿到备份验证设备的攻击者都可以通过其内置密钥覆盖原有账户，从而使合法用户彻底丢失对账户访问及控制权限。

因此，迫切需要针对工业互联网场景研究国密标准算法的ARKG构造与账户恢复协议。首先，本文提出基于SM2[16]的盲化密钥封装机制（BKEM, blinded key encapsulation mechanism）[17]，并与模糊提取器（FE, fuzzy extractor）[18]结合，设计了更安全通用的ARKG协议。与现有基于DH和KEM的ARKG相比，BKEM的引入使私钥派生必须由备份设备与服务器私钥共同参与，即使有一方是恶意的，私钥派生仍能保持安全；同时，FE将生物密钥嵌入派生私钥中，即使敌手捕获到备份验证设备，也无法恢复私钥。最后，提出了基于ARKG的账户恢复协议，在验证器设备的种子密钥生成过程中添加了密钥服务器，通过不经意伪随机函数（OPRF, oblivious pseudorandom function）[19]交互式恢复种子私钥，防止主验证设备丢失；在私钥恢复过程中，实现了服务器对用户的认证，同时，也隐式实现了用户对服务器的认证。

本文的主要工作和贡献总结如下。

1) 针对工业互联网提出了基于国密SM2的账户恢复协议，满足我国通用通信协议国产化要求的

同时，不仅有效应对主、备份验证设备丢失所引发的安全威胁，还通过引入双向认证机制，在账户恢复阶段确保服务器和用户身份真实性，提升工业互联网安全性和可靠性。

2) 作为账户恢复协议的重要组成部分，提出基于SM2的BKEM，结合模糊提取器构建新型ARKG通用框架，将生物特征嵌入私钥派生与恢复过程，并给出具体的安全性证明。

3) 对所提出的账户恢复协议的安全性分析和性能评估表明，所提协议安全性大幅增强，计算开销和通信开销仅略有增加。

1　相关工作

认证技术是工业互联网防止非法访问系统及数据的基本保障，国内外已对工业互联网场景下的身份认证协议展开大量研究。现有认证授权机制，如Radius认证协议、网络接入控制（NAC, network admission control）、自主访问控制（DAC, discretionary access control）、基于角色的访问控制（RBAC, role-based access control）等方案部署复杂、资源消耗高，难以满足工业互联网系统对可扩展性、实时性及跨平台兼容等方面的需求。因此，目前面向工业物联网（IIoT）的认证授权协议大多采用易于集成和部署、能跨平台兼容的Web认证机制。例如，Cirani等[6]对OAuth协议进行改进提出IoT-OAS方案，通过调用外部基于OAuth的授权服务实现系统的访问授权。Sciancalepore等[7]基于OAuth 2.0提出一个灵活的物联网认证和授权框架OAuth-IoT。

除OAuth外，OpenID、SAML[20]和FIDO也是目前主流的Web认证协议。Nosouhi等[5]提出在工控系统中采用W3C标准FIDO2，通过无口令认证与公钥加密提升认证安全性。结合生物识别或安全令牌的多因素认证，有效防止口令泄露、钓鱼攻击和非法访问，增强系统安全与可靠性。为进一步避免硬件令牌丢失导致的账户失效问题，Frymann等[11]提出ARKG原语，生成与种子公钥相关联的派生公钥。ARKG只需派生公钥、种子私钥和一些辅助信息，即可稍后或异步计算对应的派生私钥。使用ARKG计算的密钥对在统计分布上接近随机密钥，因此可安全用于公钥密码系统。

ARKG本身是为FIDO2的WebAuthn[21]协议应用而提出的，其设想用途是实现WebAuthn账户的恢复和授权。在WebAuthn中，验证器（如YubiKey或Windows Hello）为用户管理私钥，若验证器丢失或损坏，用户将无法访问，账户恢复成为难题。ARKG生成的不可链接公钥符合WebAuthn的强不可链接性要求，因此用户可以使用同一账户向服务器注册新的身份验证器，以在主验证设备丢失后进行账户恢复。

除账户恢复外，ARKG也被用于一种代理签名方案，来支持跨代理的不可链接性，称为“具有不可链接授权的代理签名”[22]。此类方案为委托者提供了一种通过凭证授权方法将签名权限委托给代理签名者的能力，其中代理的公钥由委托者签署。这些代理签名支持WebAuthn账户委托，ARKG作为基础组件生成的凭证具备不可链接性，其异步计算使委托过程可以是非交互式的。

最初的ARKG仅限于单个群中基于离散对数的密钥，不兼容其他密码系统，限制了其适用范围。因此Frymann等[12]提出基于配对的ARKG，并扩展至多种签名，如BLS（Boneh-Lynn-Shacham）[23]、Waters[24]、CL（Camenisch-Lysyanskaya）[25]、等价类上的结构保留签名[26]和PS（Pointcheval-Sanders）[27]，为ARKG在不同应用中的使用提供重要参考。此外，他们还提出基于格的ARKG结构[13]，来提供抗量子特性。

2　基础知识

2.1　SM2公钥加密

选定椭圆曲线系统参数$\mathrm{ECC}=(F_p,E(F_p),G,n)$、密钥派生函数$\mathrm{KDF}:\{0,1\}^*\times \mathrm{ml}\rightarrow\{0,1\}^{\mathrm{ml}}$和哈希函数$h:\{0,1\}^*\rightarrow\{0,1\}^{\mathrm{hl}}$，ml和hl分别表示待加密消息和哈希输出的长度。用户私钥为$\mathrm{SK}=d\in[1,n-1]$，对应的公钥$\mathrm{PK}=[d]G\in E(F_p)$。由于篇幅限制，对SM2公钥加密[16]的步骤简介如下。

1) $\mathrm{Enc}(\mathrm{pp},\mathrm{PK},M)\rightarrow C$。加密方随机选择$k\in[1,n-1]$，并计算$C_1=k[G]=(x_1,y_1)\in E(F_p)$，其中$x_1$和$y_1$分别表示点$C_1$的横纵坐标。用公钥PK计算椭圆曲线点$[k]\mathrm{PK}=(x_2,y_2)$，并基于该点的横纵坐标生成长度为ml的字符串：$t=\mathrm{KDF}(x_2\|y_2,\mathrm{ml})$。若生成的$t$全为0，则重新加密。计算$C_2=M\oplus t$，并用点$[k]\mathrm{PK}$的横纵坐标计算

$C_3 = h(x_2 \| M \| y_2)$。最后输出消息M的密文$C = (C_1, C_2, C_3)$。

2) Dec(pp,SK,C) → M'。解密方用私钥和密文C_1计算[SK]C_1 = [k]PK = (x_2, y_2)，并恢复字符串t = KDF($x_2 \| y_2$,ml)。如果t全为0，则解密失败直接退出。利用字符串t恢复$M' = C_2 \oplus t$，计算对应的$C_3' = h(x_2 \| M' \| y_2)$。判断$C_3'$和$C_3$是否相等，如果相等，则解密成功，否则，解密失败。

2.2 盲化密钥封装机制

相比于传统的KEM算法，Boyd等[17]提出的盲化密钥封装机制增加了盲化算法Blind和解盲算法Unblind，简介如下。

1) Setup(λ) → pp：根据选定的λ，该算法返回公共参数pp。

2) KeyGen(pp) → (ek,dk)：根据确定的pp，该算法返回封装密钥ek和解封装密钥dk。

3) Encap(pp,ek) → (C,k)：根据输入的公共参数pp和封装密钥ek，封装器执行该算法并返回封装值C和一个密钥k。

4) Blind(pp,ek,C) → ($\tilde{C}$,k')：根据封装密钥ek和封装值C，该算法输出盲化的封装值$\tilde{C}$和解盲密钥k'。

5) Decap(pp,dk,$\tilde{C}$) → $\tilde{k}$：根据解封装密钥dk和盲化的封装值$\tilde{C}$，该算法输出盲化密钥$\tilde{k}$。

6) UnBlind(pp,k',$\tilde{k}$) → k：根据解盲密钥k'和盲化密钥$\tilde{k}$，该算法输出密钥k。

2.3 模糊提取器

模糊提取器[18]可以从生物信息中提取安全密钥，允许存在一定的干扰噪声，只要生物模板的输入在一定误差范围内，就可以输出一个相同的均匀字符串。模糊提取器由一对概率多项式时间算法FE = (Gen,Rep)组成，具体如下。

1) (R,P) ← Gen(w)：输入生物特征模板w，该算法输出一个均匀字符串R和一个公开的辅助字符串P。

2) R ← Rep(P,w')：输入辅助字符串P和认证模板w'，如果Dist(w,w')不超过预定的阈值，则该算法恢复字符串R，否则输出⊥。

2.4 异步远程密钥生成

Frymann等[11]提出的ARKG由以下算法组成。

1) Setup(λ) → pp：根据选定的λ，该算法返回公共参数pp。

2) KeyGen(pp) → (sk,pk)：根据确定的pp，该算法返回种子密钥对(sk,pk)。公钥与主验证设备共享，而私钥由备份验证设备持有。

3) DerivePK(pp,pk,aux) → (pk′,cred)：根据系统公共参数pp、种子公钥pk，该算法概率性地返回pk′和绑定到输入aux的相应凭证cred。

4) DeriveSK(pp,sk,cred) → sk′：根据种子私钥sk和凭证cred，该算法确定性地返回cred对应的私钥sk′，如果凭证无效，则返回⊥。

ARKG方案安全性包含2个安全属性：公钥不可链接性和私钥安全性。公钥不可链接性要求派生的密钥对不应链接到种子公钥；私钥安全性要求在不了解种子私钥的情况下，对创建有效的凭证和派生密钥应该是不可行的。

3 系统模型和安全目标

3.1 工业互联网系统架构

如图1所示，本文将Purdue模型[28]作为整个工业互联网分层数据流的参考架构，将设备层与网络层进行分离。

在ICS中，海量复杂数据被传送到服务器计算和存储。用户访问远程站点资源时，参考Nosouhi等[5]针对ICS提出的认证方案，采用标准多因子认证协议（如FIDO2）进行用户身份验证和授权访问管理。注册用户向工业互联网终端系统发送初始访问请求，终端系统将其重定向到认证授权服务器，由该服务器通过FIDO2协议完成身份验证。成功登录后，服务器授权用户访问工业互联网资源。该认证机制与本文所提出的账户恢复协议相兼容，因此不再就认证协议进行详细描述。

3.2 账户恢复系统模型

在上述工业互联网系统架构基础上，本文抽象出账户恢复系统模型，如图2所示。该模型包括主验证设备（PAD, primary authentication device）、备份验证设备（BAD, backup authentication device）、密钥服务器（KS, key server）和认证授权服务器（AAS, authentication and authorization server）4类实体。

1) 主验证设备：用户使用PAD向AAS认证身份，从而访问工业互联网。为防止设备丢失，PAD可远程为BAD生成一个新的注册公钥并向AAS进行注册。

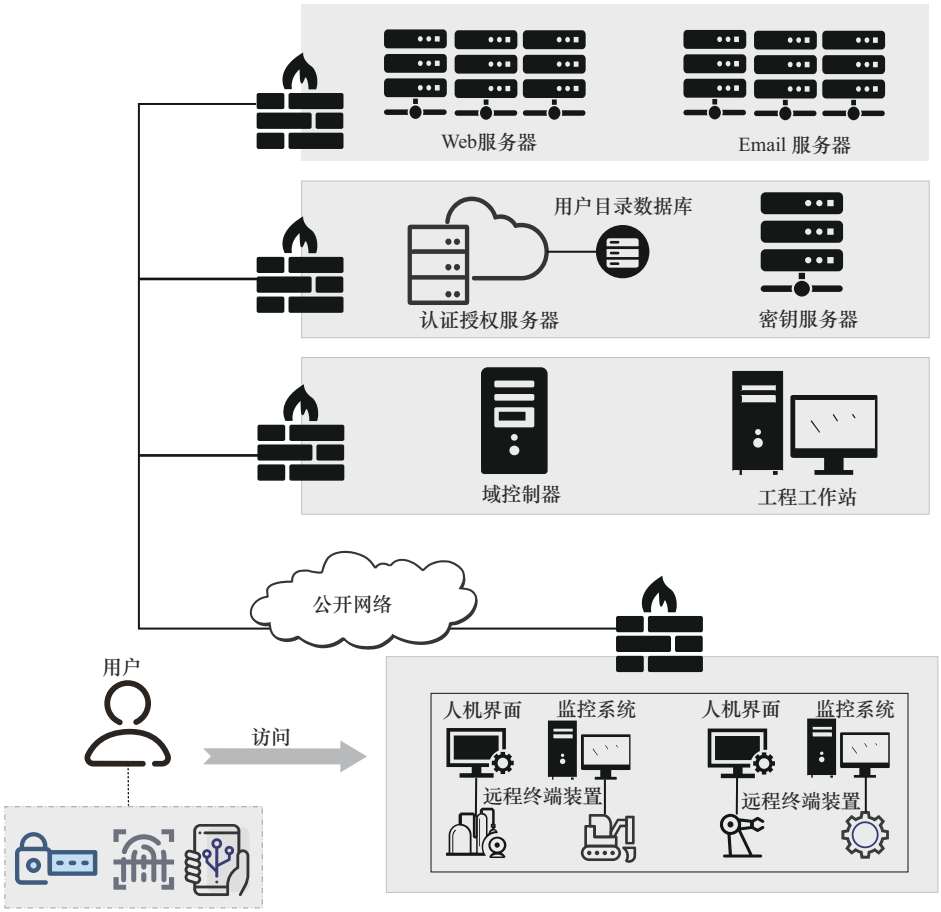

图1　工业互联网系统架构

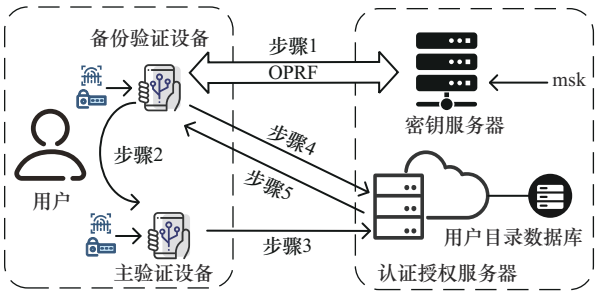

图2　账户恢复系统模型

2) 备份验证设备：当PAD丢失时，用户启用BAD与AAS交互导出注册公钥对应的私钥并进行验证，从而重新获得对账户的访问权。

3) 密钥服务器：PAD与BAD通过使用用户口令来查询KS获取其种子密钥对。

4) 认证授权服务器：AAS对持有认证设备的用户进行身份验证，并授予其对工业互联网设备的访问权；除此之外，AAS保存BAD注册公钥及其凭证信息，用于PAD丢失后的账户恢复。

BAD首先通过OPRF交互生成其种子公私钥对（步骤1），并共享给PAD（步骤2）；在BAD注册阶段，由PAD通过认证授权后，派生出BAD的注册公钥并发送给AAS进行注册（步骤3）；当用户PAD丢失，启用BAD进行账户恢复，AAS向BAD发起挑战（步骤4），BAD用种子私钥导出注册私钥对挑战签名返回AAS进行验证（步骤5），验证成功后则恢复账户。

3.3　敌手模型

假设一个概率多项式时间（PPT, probabilistic polynomial-time）敌手$\mathcal{A}$以经典方式与诚实方交互。由于备份验证设备与主验证设备之间的初始配对只发生一次，持续时间很短且在用户本地执行，不通过公共网络渠道传输信息。因此假设该过程在可信环境中执行，且敌手无法将自己随机选择的长期公钥注入用户的主验证设备中。

在备份验证设备注册过程中，主验证设备将派生公钥和账户相关恢复信息发送给服务器，假设该交互通过安全信道进行。在账户恢复过程中，用户并未向服务器进行身份认证，且没有安全信道存

在，因此，该过程允许敌手丢弃、修改或注入消息。

游戏描述 恢复认证游戏的形式化描述如图3所示。该游戏旨在评估敌手能否在账户恢复协议中，伪造有效的签名。

敌手能力 敌手$\mathcal{A}$获得公共参数pp、BAD的种子公钥$\mathrm{pk_B}$以及BAD所存储的cred作为输入，且可以访问以下预言机。

1) DerivePK：从种子公钥$\mathrm{pk_B}$和验证设备中的辅助数据aux中派生出公钥$\mathrm{pk'_B}$和恢复凭证cred。该预言机模拟了在PAD上生成的派生密钥和恢复凭证。

2) Ch-auth：生成一个随机的挑战值ch，对应于账户恢复过程中AAS发送的挑战。

3) Sign：使用恢复凭证cred派生出私钥$\mathrm{sk'_B}$，并用该派生私钥对消息m签名生成σ。

4) LeakSK：模拟派生私钥的泄露，敌手可以提供恢复凭证cred（BAD丢失攻击），预言机返回DeriveSK的输出，如果派生成功则返回$\mathrm{sk'_B}$，否则返回⊥。

3.4 安全目标

从加密原语的角度看，ARKG用于从种子公钥派生出新公钥，且派生私钥仅可由种子私钥恢复。因此Frymann等[11]将ARKG的核心安全属性描述为在未知长期私钥的情况下，敌手无法派生有效密钥对。但其ARKG所描述的加密核心忽略了敌手在协议执行过程中的实际视角，因此并未涵盖协议所需的实际安全属性。在本文的工业互联网账户恢复场景中，关键安全目标是即使敌手掌握部分信息，也无法对挑战值生成有效签名，从而保证协议安全。因此，本节以敌手能否成功恢复认证的能力来抽象密钥安全性，能更准确反映ARKG在工业互联网中的使用情况。基于ARKG的账户恢复协议需要满足以下几点安全目标。

1) 私钥保密性：由于BAD的种子公钥可能由PAD进行多次派生并在不同的AAS进行注册，因此，必须保证PAD在丢失之前注册所有恢复种子私钥的保密性。

2) 公钥不可链接性：PAD为BAD多次派生的注册公钥之间需确保不可链接性，即AAS无法确定它们是为相同还是不同的BA注册的。

3) 恢复认证安全[15]：在账户恢复过程中，即使敌手知道了一些信息，也无法对给定的挑战值生成有效的签名，从而保证恢复协议的认证安全性。

4) 抵御验证器丢失攻击：任何不持有用户口令及生物特征的敌手获取到BAD后都无法恢复账户。

4 工业互联网账户恢复协议

4.1 设计思想

本文旨在构建一个安全可靠的工业互联网账

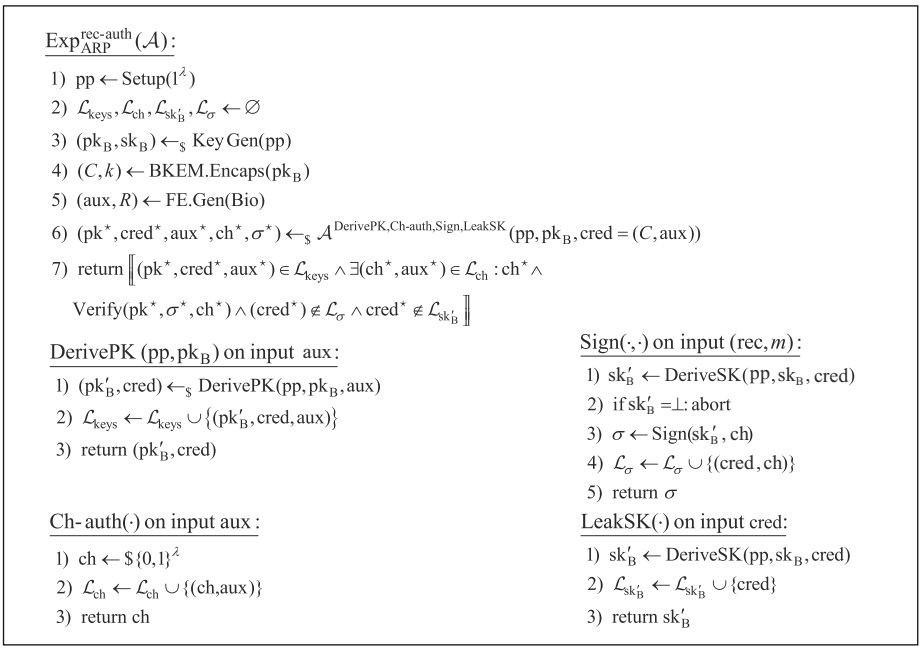

图3 恢复认证游戏的形式化描述

户恢复协议，主要创新点体现在如下 3 个方面。第一，针对现有协议不符合国密标准及缺乏双向认证的问题，基于 SM2 设计 BKEM，并构造由服务器与备份设备协同派生私钥的 ARKG，既降低单方密钥生成的信任风险，又隐式实现双向认证。第二，针对备份验证设备丢失攻击，结合 FE 将生物特征嵌入 ARKG 密钥派生，使敌手即使获得设备，在未掌握生物特征的情况下也无法重构私钥。最后，为防止主验证设备丢失，引入 OPRF 交互式恢复种子私钥，避免其明文存储，形成设备丢失防护与密钥安全的双重保障。综上所述，所提方案在满足国密合规性的同时提升了认证安全与设备丢失防御能力，为工业互联网账户恢复提供了新的技术途径。

4.2　基于 SM2 的 BKEM

基于 SM2 的 BKEM 算法构造如图 4 所示。

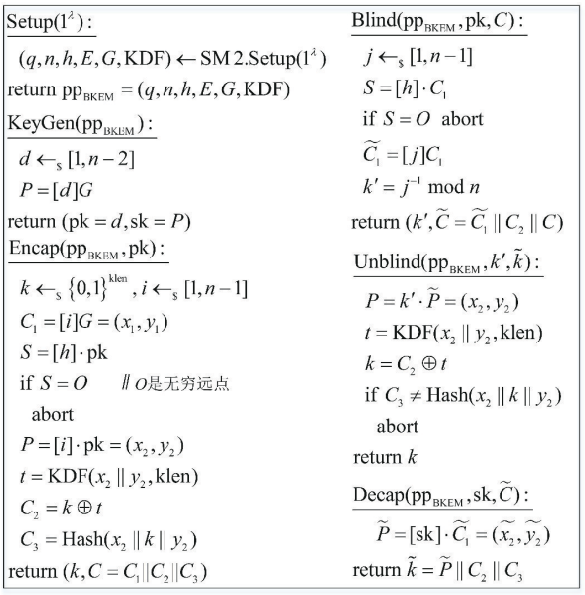

图 4　基于 SM2 的 BKEM 算法构造

1) Setup (1^λ) → pp：系统初始化算法，输入系统安全参数，执行 SM 2.Setup (λ)，输出公共参数 $\mathrm{pp_{BKEM}} = (q,n,h,E,G,\mathrm{KDF})$。

2) KeyGen ($\mathrm{pp_{BKEM}}$) → (pk,sk)：输入公共参数 params，用户随机选取秘密值 $d \in [1,n-2]$ 作为私钥，计算公钥 $P = [d]G$，输出公私钥对 $(\mathrm{pk},\mathrm{sk}) = (P,d)$。

3) Encap ($\mathrm{pp_{BKEM}}$,pk) → (k,C)：选择一个随机密钥 $k \in \{0,1\}^{\mathrm{klen}}$ 和随机数 $i \in [1,n-1]$，计算椭圆曲线点 $C_1 = [i]G = (x_1,y_1)$ 和 $S = [h] \cdot \mathrm{pk}$，若 S 是无穷远点，则报错并退出；计算点 $P = [i] \cdot \mathrm{pk} = (x_2,y_2)$、$t = \mathrm{KDF}(x_2 \| y_2, \mathrm{klen})$、$C_2 = k \oplus t$ 及 $C_3 = \mathrm{Hash}(x_2 \| k \| y_2)$，输出 $(k, C = C_1 \| C_2 \| C_3)$。

4) Blind ($\mathrm{pp_{BKEM}}$,pk,C) → ($\tilde{C}$,k')：选择随机数 $j \in [1,n-1]$，从 C 中提取 C_1 验证是否满足椭圆曲线方程，不满足则退出；计算椭圆曲线点 $S = [h] \cdot C_1$，若 S 是无穷远点，则报错并退出；计算 $\widetilde{C_1} = [j]C_1$、$k' = j^{-1} \bmod n$ 及 $\tilde{C} = \widetilde{C_1} \| C_2 \| C_3$，输出 $(k', \tilde{C})$。

5) Decap ($\mathrm{pp_{BKEM}}$,sk,$\tilde{C}$) → $\tilde{k}$：从 $\tilde{C}$ 中取出比特串 $\widetilde{C_1}$，计算椭圆曲线点 $\tilde{P} = [\mathrm{sk}] \cdot \widetilde{C_1} = (\widetilde{x_2}, \widetilde{y_2})$，输出 $\tilde{k} = \tilde{P} \| C_2 \| C_3$。

6) Unblind ($\mathrm{pp_{BKEM}}$,k',$\tilde{k}$) → k：从 $\tilde{k}$ 中提取 $\tilde{P}$，计算 $P = k' \cdot \tilde{P} = (x_2,y_2)$、$t = \mathrm{KDF}(x_2 \| y_2, \mathrm{klen})$ 及 $k = C_2 \oplus t$，验证 $C_3 \stackrel{?}{=} \mathrm{Hash}(x_2 \| k \| y_2)$，若不相等，则报错并退出；若相等，则输出 k。

安全性：BKEM 安全性证明见附录 1。

4.3　基于 BKEM 的 ARKG

本文采用基于 SM2 的 BKEM 来构造 ARKG，允许用户通过认证服务器动态派生公私钥对。在设置算法中，执行 BKEM . Setup (1^λ) 输出系统公共参数 $\mathrm{pp} = (q,n,h,E,G,\mathrm{KDF})$，后续认为 pp 为默认输入；密钥生成算法也与 SM 2.KeyGen 保持一致，生成 SM2 密钥对 $(\mathrm{sk} = d \in [1,n-2], \mathrm{pk} = [d]G)$。

在公钥派生算法中，将 BKEM 封装得到的密钥材料与由 FE 生成的用户生物密钥组合，通过伪随机函数导出密钥种子并计算对应的公钥。具体如算法 1 所示。

算法 1　DerivePK (pk,aux = ($\mathrm{pk_S}$,Bio))

1) 输入：种子公钥 pk 以及服务器公钥 $\mathrm{pk_S}$ 和生物特征 Bio。

2) BKEM(C,k) ← BKEM .Encap ($\mathrm{pk_S}$)；//执行 BKEM 封装算法，其中，$\mathrm{pk_S}$ 为认证服务器公钥，k 为第一个密钥材料。

3) 计算 (R,P) ← FE . Gen (Bio)；//执行模糊提取器密钥生成算法，其中，R 为第二个密钥材料，P 为辅助数据。

4) 计算 $\tau \leftarrow \mathrm{PRF}(k,R)$；//以 k 和 R 为输入通过不经意伪随机函数计算密钥种子。

5) 计算 $\mathrm{pk}' \leftarrow [\tau] \cdot G + \mathrm{pk}$。

6) 输出：派生公钥 pk' 及凭证 $\mathrm{cred} = (C,P)$。

在私钥派生算法中，需要用户和服务器交互执行，本文将该过程分为3个子算法：首先由用户执行 $\mathrm{UserBegin}(\mathrm{pk_S},C) \rightarrow (\tilde{C},k')$，根据 BKEM 盲化算法计算封装 C 的盲化值 $\tilde{C}$ 及对应的解盲密钥 k'，并盲化值 $\tilde{C}$ 发送给认证服务器；其次由服务器执行 $\mathrm{Server}(\mathrm{sk_S},\tilde{C}) \rightarrow \tilde{k}$，用私钥解封装得到盲化的密钥 $\tilde{k}$ 并返回给用户；最后用户执行 $\mathrm{UserComplete}(\mathrm{sk},P,\tilde{k},\mathrm{Bio}') \rightarrow \mathrm{sk}'$ 导出对应的私钥。具体如算法2所示。

算法2　$\mathrm{DeriveSK}(U(\mathrm{aux},\mathrm{cred},\mathrm{sk}) \leftrightarrow S(\mathrm{sk_S}))$

1) 用户输入服务器公钥 $\mathrm{pk_S}$ 凭证中的密文 C，执行 $\mathrm{UserBegin}(\mathrm{pk_S},C) \rightarrow (\tilde{C},k')$，即计算 $(\tilde{C},k') \leftarrow \mathrm{BKEM.Blind}(\mathrm{pk_S},C)$，发送 $\tilde{C}$ 给服务器。

2) 服务器执行 $\mathrm{Server}(\mathrm{sk_S},\tilde{C}) \rightarrow \tilde{k}$，即计算 $\tilde{k} \leftarrow \mathrm{BKEM.Decap}(\mathrm{sk_S},\tilde{C})$，并返回 $\tilde{k}$ 给用户。

3) 用户执行 $\mathrm{UserComplete}(\mathrm{sk},P,\tilde{k},\mathrm{Bio}') \rightarrow \mathrm{sk}'$，具体步骤如下。

① 计算 $k \leftarrow \mathrm{BKEM.Unblind}(k',\tilde{k})$；//执行BKEM去盲算法，恢复第一个密钥材料 k。

② 计算 $R \leftarrow \mathrm{FE.Rep}(\mathrm{Bio}',P)$；//执行模糊提取器密钥重构算法，恢复第二个密钥材料 R。

③ 计算 $\tau \leftarrow \mathrm{PRF}(k,R)$；//以 k 和 R 为输入通过不经意伪随机函数计算密钥种子。

④ 计算 $\mathrm{sk}' \leftarrow \tau + \mathrm{sk}$；

4) 输出 sk'。

安全性：ARKG安全性证明见附录2。

4.4　基于SM2 ARKG的账户恢复协议

工业互联网中基于ARKG的账户恢复协议可分为4个主要阶段：系统初始化阶段、设置阶段、注册阶段与恢复阶段。在设置阶段中，BAD与KS交互获取其种子公私钥对，并将种子公钥共享给PAD。在注册阶段中，PAD使用该种子公钥代表BAD获取新的公钥用于向AAS进行注册，且该注册公钥对应的私钥仅由BAD持有，用于后续的账户恢复。当PAD丢失或失效后进入恢复阶段，用户使用BAD与AAS交互恢复注册公钥对应的私钥并进行验证，恢复用户的访问权限。协议相关符号说明如表1所示，具体细节如图5所示。

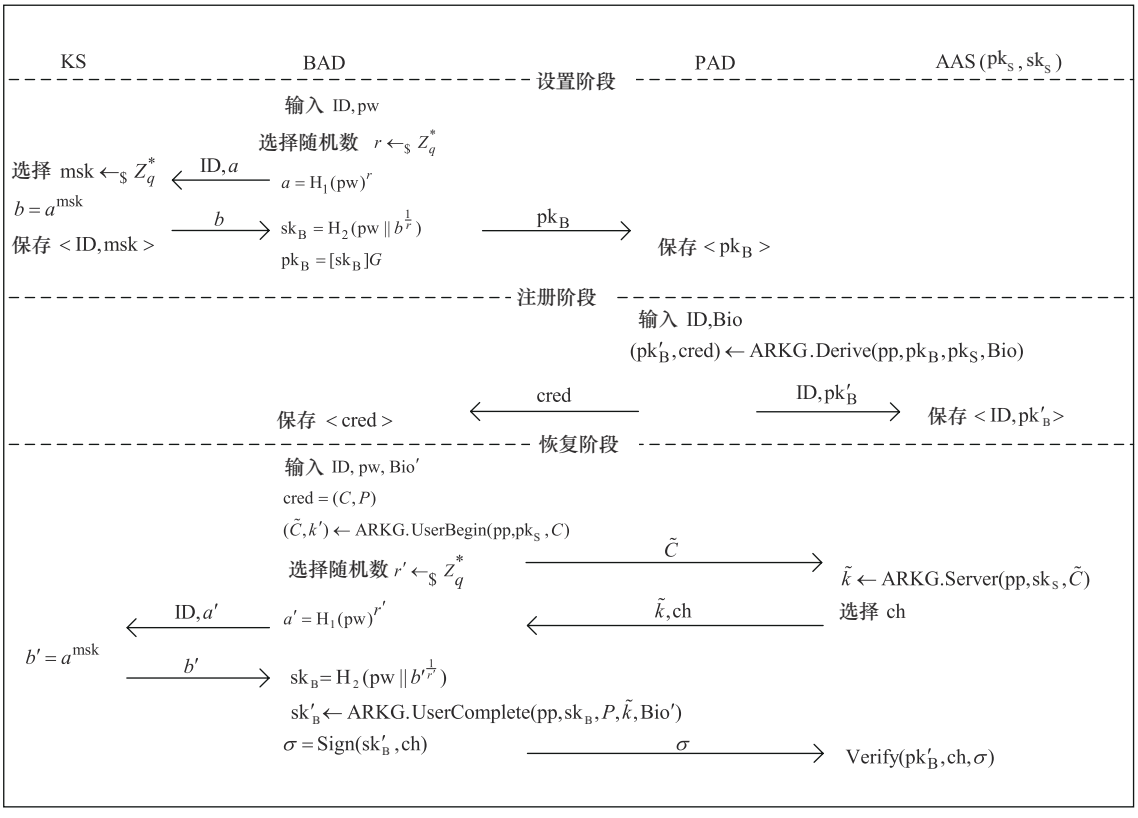

图5　账户恢复协议各阶段步骤

表1　　符号说明

符号	含义
ID,pw,Bio	用户身份标识、口令及生物特征
$\mathrm{pk_S},\mathrm{sk_S}$	AAS的公、私钥对
$\mathrm{pk_B},\mathrm{sk_B}$	BAD的种子公、私钥对
$\mathrm{pk'_B},\mathrm{sk'_B}$	BAD的派生公、私钥对
msk	OPRF密钥
Sign,Verify	SM2签名、验签函数
$\mathrm{H_1},\mathrm{H_2}$	单向哈希函数

4.4.1　系统初始化阶段

在系统初始化阶段，AAS生成所有的系统参数，详细步骤如下。

1) AAS生成SM2公钥密码算法所需的参数。AAS确定有限域F_p及对应的椭圆曲线$E(F_p)$，随机确定一个基点$G=(x_G,y_G)$，基点G的阶为素数q。

2) AAS选择哈希函数$\mathrm{H_1}:\{0,1\}^* \to Z_q^*$，以及$\mathrm{H_2}:\{0,1\}^* \to Z_p^*$。

3) AAS选择随机数$\mathrm{sk_S} \in [1,n-1]$作为AAS的私钥，并计算对应的公钥为$\mathrm{pk_S}=[\mathrm{sk_S}]G$，然后AAS保存$\mathrm{sk_S}$并将参数$\left\{F_q,E(F_q),G,n,\mathrm{KDF_1},\mathrm{KDF_2},\mathrm{H}\right\}$公开给所有用户。

4.4.2　设置阶段

BAD与KS采用OPRF协议进行交互生成种子密钥对，并将公钥共享给PAD。本文假设此阶段是受信任的，因为公钥共享过程仅由2个验证设备在用户本地执行，不涉及与认证授权服务器的交互。详细步骤如下。

1) 用户向BAD输入口令pw，选择随机数$r \leftarrow {}_{\$}Z_q$，计算$a=\mathrm{H_1}(\mathrm{pw})^r$，将$a$发送给KS。

2) 密钥服务器持有OPRF密钥msk，收到a后，计算$b=a^{\mathrm{msk}}$，将b返回给BAD。

3) BAD收到消息后，计算$\mathrm{sk_B}=\mathrm{H_2}(\mathrm{pw}\|b^{\frac{1}{r}})$作为BAD的种子私钥，并计算公钥$\mathrm{pk_B}=[\mathrm{sk_B}]G$共享给PAD，备份验证设备BAD初始化完成。

4.4.3　注册阶段

当PAD为BAD远程生成公钥并向AAS进行注册时，假设PAD已在AAS中注册过合法账户，并由ID进行标识。PAD将为完成设置阶段的BAD派生新的公钥并发送给AAS注册：输入系统公共参数、AAS公钥$\mathrm{pk_S}$、BAD公钥$\mathrm{pk_B}$以及用户生物特征Bio，执行DerivePK(pk,aux=($\mathrm{pk_S}$,Bio))算法，将(ID,$\mathrm{pk'_B}$)发送给AAS进行注册，将cred发送给BAD表示备份注册完成。

4.4.4　恢复阶段

当PAD丢失或失效时，BAD向AAS发出账户恢复请求，用户在AAS上提供账户标识符ID，然后可以检索与该账户关联的凭证返回给BAD进行私钥恢复，并向AAS进行身份验证以重新获取对账户的访问权限。具体步骤如下。

1) BAD将cred解析为(C,P)，用户输入标识ID、口令pw以及新采样的生物特征Bio′，执行UserBegin($\mathrm{pk_S}$,C)$\to(\tilde{C},k')$，选择随机数$r' \leftarrow {}_{\$}Z_q$计算$a'=\mathrm{H_1}(\mathrm{pw})^{r'}$，将$a'$发送给KS，并将盲化的密钥封装值$\tilde{C}$发送给AAS进行解封装。

2) AAS执行Server($\mathrm{sk_S}$,$\tilde{C}$)$\to\tilde{k}$，并随机生成一个挑战ch与盲化密钥$\tilde{k}$一起返回BAD。

3) KS计算$b'=a'^{\mathrm{msk}}$，将b'返回给BAD。

4) BAD收到消息后，计算$\mathrm{sk_B}=\mathrm{H_2}(\mathrm{pw}\|b'^{\frac{1}{r'}})$，执行UserComplete($\mathrm{sk_B}$,$P$,$\tilde{k}$,Bio′)$\to\mathrm{sk'_B}$，BAD用导出的私钥对挑战签名，计算$\sigma=\mathrm{Sign}(\mathrm{sk'_B},\mathrm{ch})$，将签名$\sigma$返回给AAS进行验证。

5) AAS用BAD的注册公钥验证Verify($\mathrm{pk'_B}$,σ)，输出1则账户恢复成功，输出0则恢复失败。

5　安全性分析

5.1　安全性证明

定义1　恢复认证安全。设ARKG=(Setup,KeyGen,DerivePK,DeriveSK)是一个异步远程密钥生成方案，如果任意多项式时间敌手$\mathcal{A}$赢得游戏$\mathrm{Exp}_{\mathrm{ARP}}^{\mathrm{rec-auth}}(\mathcal{A})$的优势

$$\mathrm{Adv}_{\mathrm{ARP}}^{\mathrm{rec-auth}}(\mathcal{A}):=\left|\Pr\left[\mathrm{Exp}_{\mathrm{ARP}}^{\mathrm{rec-auth}}(\mathcal{A})=1\right]\right| \quad (1)$$

是可忽略的，则基于ARKG的账户恢复协议是满足恢复认证安全的。

定理1　设BKEM是第5.1节基于SM2的盲化密钥封装实例，满足IND安全性。设Sig是SM2签名算法，满足存在性不可伪造（EUF-CMA, existential unforgeability under adaptively chosen-message attack）[29]。设KDF是建模为PRF的安全密钥

派生函数，FE是模糊提取方案实例，OPRF是基于2HashDH[30]的实例。对于任意针对恢复认证安全性的多项式时间敌手$\mathcal{A}$，都存在与$\mathcal{A}$运行时间大致相同的PPT算法$\mathcal{B}_0$,$\mathcal{B}_1$,$\mathcal{B}_2$,$\mathcal{B}_3$和$\mathcal{B}_4$，可以进行最多q次查询，使

$$\mathrm{Adv}_{\mathrm{ARP}}^{\mathrm{rec\text{-}auth}}(\mathcal{A})\leqslant q\cdot(\mathrm{Adv}_{\mathrm{SM2.BKEM}}^{\mathrm{IND}}(\mathcal{B}_0)+\mathrm{Adv}_{\mathrm{KDF}}^{\mathrm{PRF}}(\mathcal{B}_1)+\mathrm{Adv}_{\mathrm{FE}}(\mathcal{B}_2)+\mathrm{Adv}_{\mathrm{OPRF}}(\mathcal{B}_3)+\mathrm{Adv}_{\mathrm{SM2.Sig}}^{\mathrm{EUF\text{-}CMA}}(\mathcal{B}_4))+\varepsilon \tag{2}$$

证明

Game_0：第一个游戏是$\mathrm{Exp}_{\mathrm{ARP}}^{\mathrm{rec\text{-}auth}}(\mathcal{A})$实验。对于每个$\mathrm{Game}_i$，设$E_i$表示在该游戏中敌手成功猜测的事件，根据rec-auth优势定义，有

$$\mathrm{Adv}_{\mathrm{ARP}}^{\mathrm{rec\text{-}auth}}(\mathcal{A})=2\left|\Pr[E_0]-\frac{1}{2}\right| \tag{3}$$

Game_1：该游戏与Game_0的区别在于修改了预言机DerivePK的行为，将原本传递给KDF的BKEM密文替换为等长的随机值。如果存在一个有效的敌手$\mathcal{A}$能够区分Game_0和Game_1，则意味着该敌手可以攻击BKEM的IND安全性。PPT算法$\mathcal{B}_0$接收BKEM挑战三元组$(\mathrm{pk}^\star,k^\star,c^\star)$，使用$\mathrm{pk}^\star$初始化$\mathrm{Exp}_{\mathrm{ARP}}^{\mathrm{rec\text{-}auth}}(\mathcal{A})$作为$\mathrm{pk}_{\mathrm{B}}$。然后$\mathcal{B}_0$在DerivePK中使用挑战公钥$\mathrm{pk}^\star$进行封装，并使用挑战密钥$k^\star$作为输入传递给KDF。密文的输出则被替换为挑战密文$c^\star$作为恢复凭证的组成部分。

为了模拟签名预言机，$\mathcal{B}_0$存储一个派生密钥的列表，这些密钥在DerivePK中生成但在正常操作中被丢弃。Ch-auth可以模拟该行为，因为它没有任何秘密输入。最后，敌手$\mathcal{A}$结束并输出猜测b，$\mathcal{B}_0$则将该猜测作为它对BKEM挑战者的回答。因此，Game_0与Game_1之间的差异取决于BKEM的IND安全性。对于一个有效的多项式敌手$\mathcal{B}_0$，可以进行最多q次查询，有

$$\left|\Pr[E_1]-\Pr[E_0]\right|\leqslant q\cdot\mathrm{Adv}_{\mathrm{SM2.BKEM}}^{\mathrm{IND}}(\mathcal{B}_0) \tag{4}$$

Game_2：该游戏与Game_1的区别在于进一步修改了预言机DerivePK的执行过程，将KDF的输出τ替换为等长的随机值。如果存在一个有效的敌手$\mathcal{A}$能够区分Game_1和Game_2，则意味着该敌手可以攻击底层的KDF安全性。本文将KDF的安全性建模为一个伪随机函数PRF。PPT算法$\mathcal{B}_1$按照指定行为为敌手$\mathcal{A}$初始化$\mathrm{Exp}_{\mathrm{ARP}}^{\mathrm{rec\text{-}auth}}(\mathcal{A})$，但没有直接调用DerivePK中的密钥派生函数，而是将输入转发给由PRF挑战者提供的PRF预言机。其他预言机的模拟与Game_1完全相同。最后，敌手$\mathcal{A}$结束并输出猜测b，$\mathcal{B}_1$则将该猜测作为它对PRF挑战者的回答。因此，Game_1与Game_2之间的差异取决于PRF的安全性。对于一个有效的多项式敌手$\mathcal{B}_1$，可以进行最多q次查询，有

$$\left|\Pr[E_2]-\Pr[E_1]\right|\leqslant q\cdot\mathrm{Adv}_{\mathrm{KDF}}^{\mathrm{PRF}}(\mathcal{B}_1) \tag{5}$$

Game_3：该游戏与Game_2的区别在于继续修改了预言机DerivePK的执行过程，将FE输出的生物密钥R替换为等长的随机值。如果存在一个有效的敌手$\mathcal{A}$能够区分Game_3和Game_2，则意味着该敌手可以攻击底层的FE安全性。PPT算法$\mathcal{B}_2$按照指定行为为敌手$\mathcal{A}$初始化$\mathrm{Exp}_{\mathrm{ARP}}^{\mathrm{rec\text{-}auth}}(\mathcal{A})$，但没有直接调用DerivePK中的模糊提取器密钥重构算法，而是将输入转发给由FE挑战者提供的FE预言机。其他预言机的模拟与Game_2完全相同。最后，敌手$\mathcal{A}$结束并输出猜测b，$\mathcal{B}_2$则将该猜测作为它对FE挑战者的回答。因此，Game_3与Game_2之间的差异取决于FE的安全性。对于一个有效的多项式敌手$\mathcal{B}_2$，可以进行最多q次查询，有

$$\left|\Pr[E_3]-\Pr[E_2]\right|\leqslant q\cdot\mathrm{Adv}_{\mathrm{FE}}(\mathcal{B}_2) \tag{6}$$

Game_4：该游戏与Game_3的区别在于预言机DeriveSK的行为，将输入sk_{B}替换为等长的随机值。如果存在一个有效的敌手$\mathcal{A}$能够区分Game_4和Game_3，则意味着该敌手可以攻击底层的OPRF安全性。PPT算法$\mathcal{B}_3$按照指定行为为敌手$\mathcal{A}$初始化$\mathrm{Exp}_{\mathrm{ARP}}^{\mathrm{rec\text{-}auth}}(\mathcal{A})$，但没有直接调用DeriveSK，而是将OPRF挑战者发送的挑战私钥作为输入。其他预言机的模拟与Game_3完全相同。最后，敌手$\mathcal{A}$结束并输出猜测b，$\mathcal{B}_3$则将该猜测作为它对OPRF挑战者的回答。因此，Game_4与Game_3之间的差异取决于OPRF的安全性。对于一个有效的多项式敌手$\mathcal{B}_3$可以进行最多q次查询，有

$$\left|\Pr[E_4]-\Pr[E_3]\right|\leqslant q\cdot\mathrm{Adv}_{\mathrm{OPRF}}(\mathcal{B}_3) \tag{7}$$

Game_5：该游戏与Game_4的区别在于修改了签名预言机Sign的执行过程。如果存在一个有效的敌手$\mathcal{A}$能够区分Game_4和Game_5，则意味着该敌手可以攻击SM2.Sign的EUF-CMA安全性。PPT算法$\mathcal{B}_4$按照指定行为初始化$\mathrm{Exp}_{\mathrm{ARP}}^{\mathrm{rec\text{-}auth}}(\mathcal{A})$，且持有备份验证设备的种子密钥对$(\mathrm{pk}_{\mathrm{B}},\mathrm{sk}_{\mathrm{B}})$。$\mathcal{B}_4$接收EUF-CMA挑战者发送的挑战公钥$\mathrm{pk}^\star$签名预言机Sign，

并将DerivePK输出公钥替换为挑战公钥$\mathrm{pk}^{\star}$，这意味着恢复凭证$\mathrm{cred}^{\star}$也会与$\mathrm{pk}^{\star}$关联。

对于敌手$\mathcal{A}$提出的针对恢复凭证$\mathrm{cred}^{\star}$的签名请求，$\mathcal{B}_4$会将请求转发给EUF-CMA游戏的外部签名预言机。由于DeriveSK是确定的，攻击中的签名预言机会回复出与$\mathrm{pk}^{\star}$相对应的私钥，因此使用外部签名预言机是有效的。对于其他值的签名请求，$\mathcal{B}_4$可以利用sk_{B}生成相应的签名。最后，敌手$\mathcal{A}$会输出一组值$(\mathrm{pk}^{\star},\mathrm{cred}^{\star},\mathrm{aux}^{\star},\mathrm{ch}^{\star},\sigma^{\star})$，其中，$\sigma^{\star}$是对挑战公钥$\mathrm{pk}^{\star}$和挑战$\mathrm{ch}^{\star}$的一个有效签名。由于唯一转发给$\mathcal{B}_4$外部签名预言机的查询是针对$\mathrm{cred}^{\star}$的，敌手$\mathcal{A}$只有当$(\mathrm{cred}^{\star},\mathrm{ch}^{\star})$不在已签名对列表$\mathcal{L}_{\sigma}$中时才能获胜，因此可以推断$\mathrm{ch}^{\star}$之前在$\mathcal{B}_4$的攻击中并没有被签名过。所以$\mathcal{B}_4$可以直接输出消息-签名对$(\mathrm{ch}^{\star},\sigma^{\star})$作为有效伪造。最后，敌手$\mathcal{A}$结束并输出猜测$b$，$\mathcal{B}_4$则将该猜测作为它对EUF-CMA挑战者的回答。因此，Game_4与Game_5之间的差异取决于SM2.Sign的安全性。对于一个有效的多项式敌手$\mathcal{B}_4$，可以进行最多q次查询，有

$$\left|\Pr\left[E_5\right]-\Pr\left[E_4\right]\right|\leqslant q\cdot\mathrm{Adv}_{\mathrm{SM2.Sig}}^{\mathrm{EUF\text{-}CMA}}(\mathcal{B}_4) \tag{8}$$

Game_6：该游戏与Game_5的区别在于显示模拟设备丢失攻击，敌手$\mathcal{A}$可以访问验证设备上存储的恢复凭证cred。然而，cred中的C是BKEM密文，其安全性依赖于BKEM的IND性；P是生物特征辅助数据，其安全性依赖于模糊提取器。如果敌手$\mathcal{A}$在该游戏中以不可忽略的优势获胜，则必须攻破BKEM或FE安全性。在之前的游戏Game_1和Game_3中，已将这些组件的输出进行分析，并证明敌手无法区分，因此提供C和P不显著增加敌手优势。因此该游戏与Game_5的差异可忽略不计。

$$\left|\Pr\left[E_6\right]-\Pr\left[E_5\right]\right|\leqslant\varepsilon \tag{9}$$

其中，ε依赖于BKEM和FE的安全性，可忽略。

综合以上游戏的分析，敌手在$\mathrm{Exp}_{\mathrm{ARP}}^{\mathrm{rec\text{-}auth}}(\mathcal{A})$中的优势为

$$\begin{aligned}\mathrm{Adv}_{\mathrm{ARP}}^{\mathrm{rec\text{-}auth}}(\mathcal{A})\leqslant q\cdot(&\mathrm{Adv}_{\mathrm{SM2.BKEM}}^{\mathrm{IND}}(\mathcal{B}_0)+\mathrm{Adv}_{\mathrm{KDF}}^{\mathrm{PRF}}(\mathcal{B}_1)+\\&\mathrm{Adv}_{\mathrm{FE}}(\mathcal{B}_2)+\mathrm{Adv}_{\mathrm{OPRF}}(\mathcal{B}_3)+\mathrm{Adv}_{\mathrm{SM2.Sig}}^{\mathrm{EUF\text{-}CMA}}(\mathcal{B}_4))+\varepsilon\end{aligned} \tag{10}$$

5.2　安全性对比

本节将基于SM2 ARKG的账户恢复协议与Frymann等[11]协议进行安全性对比，以证明所提协议在安全性上的优势。现有工作中，除文献[11]外，其他账户恢复协议[13-15]均聚焦于后量子场景，因此本文暂不予考虑。

如表2所示，Frymann等[11]协议与所提协议均满足私钥保密性、公钥不可链接性（证明过程见附录2）和恢复认证安全。

表2　　安全性对比

安全目标	文献[11]协议	所提协议
私钥保密性	√	√
公钥不可链接性	√	√
恢复认证	√	√
抗主验证设备丢失	√	√
抗备份验证设备丢失	×	√
双向认证	×	√

为应对主验证设备丢失，账户恢复协议使用备份设备存储派生密钥信息，但这也扩大了攻击面。Frymann等[11]将账户私钥存于主验证设备，一旦设备被窃取，敌手即可直接获取私钥。主验证设备丢失后，用户必须立即启用备份验证设备覆盖账户，使原设备失效。备份验证设备的私钥恢复完全依赖设备，任何窃取设备的敌手都能伪造用户，难以抵抗备份验证设备丢失攻击。此外，该协议仅支持服务器认证用户，缺少用户对服务器的验证，降低了协议整体的认证安全性。

本文所提协议不向主验证设备存储账户私钥，用户通过OPRF与口令恢复高熵私钥。因此窃取到主验证设备的敌手在未知口令的情况下无法从中获取任何私钥信息。主验证设备丢失后，用户可立即启用备份验证设备恢复账户控制权。对于备份验证设备，将用户生物特征密钥嵌入其注册密钥中，恢复过程必须有用户生物特征参与。因此敌手即使窃取设备，若无生物信息也无法恢复账户。此外，本文方案通过BKEM使服务器参与用户私钥的派生过程，隐式完成了用户对服务器的身份认证。

6　性能评估

6.1　计算开销分析

本文采用Python3编程语言在SM2标准中推荐的椭圆曲线SM2-P-256上实现了基于SM2的BKEM原语，并以Liberati等[31]的xlock作为模糊提取器实例，基于标准HKDF[32]与SHA-256[33]一起使用作为KDF函数实例来实现ARKG方案。本文

分别在Raspberry Pi 4 Model B 8 GB和i9-13900K 24Core 3 GHz 128 GB运行内存的环境下模拟工业互联网客户端及服务器，对ARKG及账户恢复协议进行了性能测试。

6.1.1 基于SM2的BKEM及ARKG计算开销

本文分别在客户端及服务器设备上运行基于SM2的BKEM算法各50次，统计具体算法的平均计算时间，结果如表3所示。为进行对比分析，本文对Frymann等[11]提出的基于ECDSA及文献[12]中对PS、SPSEQ（structure-preserving signatures on equivalence）、Waters这3种签名构造的ARKG方案在相同设备下进行测试。由于在账户恢复协议中，公私钥的派生通过用户的验证设备执行，因此以上现有ARKG方案均在Raspberry Pi上执行，对比数据如表4所示。

表3 基于SM2的BKEM计算开销

算法	客户端/ms	服务器/ms
BKEM.Encaps	115.69	26.87
BKEM.Blind	52.68	13.15
BKEM.Decaps	51.85	12.41
BKEM.UnBlind	55.45	13.79

表4 ARKG计算开销对比

算法	PKD/ms	SKD/ms
ECDSA-ARKG	89.19	31.28
PS-ARKG	52.44	25.80
SPSEQ-ARKG	127.50	53.36
Waters-ARKG	139.59	53.38
SM2BKEM-ARKG	166.62	119.44

与现有ARKG构造相比，本文提出的基于SM2的ARKG额外添加了FE构造，并使用BKEM生成密钥材料来增强方案安全性，因此在计算开销上有所增加。

6.1.2 账户恢复协议计算开销

文献[12]中仅提出基于PS、SPSEQ、Waters的ARKG构造，没有提出账户恢复协议，因此仅对文献[11]和所提协议进行评估。文献[11]中，客户端运算为ARKG的公私钥派生与签名生成，计算开销为123.30 ms，服务器端仅涉及一次验签运算，计算开销为0.61 ms。本文所提账户恢复协议的ARKG添加了FE和BKEM盲化与解盲运算，其客户端计算开销为286.88 ms，服务器端计算开销为12.47 ms，以增加计算开销为代价来提供抗验证设备丢失攻击及双向认证等安全特性。

6.2 存储与通信开销分析

图6为文献[11]协议与所提协议的存储和通信开销对比。按照实验设置，设哈希函数输出长度为256 bit，ECDSA与SM2均选择P-256标准椭圆曲线，用户及服务器身份长度为32 bit。

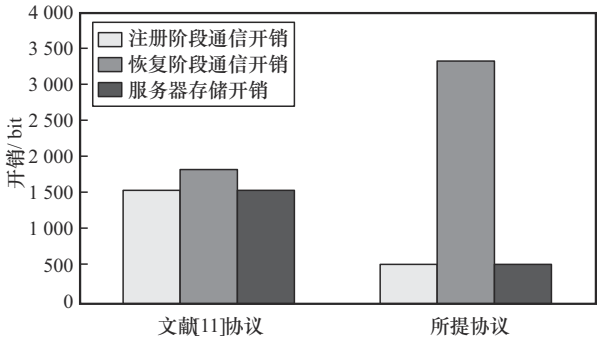

图6 存储和通信开销对比

文献[11]协议中，在注册阶段，主验证设备向服务器发送消息$\langle \mathrm{ID_C},P,\mathrm{cred}=(E,\mathrm{aux},\mu)\rangle$并保存，其中，$P$和$E$为ECDSA公钥，长度为512 bit，辅助数据aux是服务器标识符的哈希，长度为256 bit，μ是对密钥材料的MAC哈希，长度为256 bit，共$512\times2+256\times2+32=1\,568$ bit。在恢复阶段，备份验证设备与服务器之间的通信消息为$\langle \mathrm{ID_C},\mathrm{cred}=(E,\mathrm{aux},\mu),\mathrm{ID_S},\mathrm{ch},\sigma\rangle$，其中，ch是长度为256 bit的挑战，$\sigma$是SM2签名，长度为512 bit，共$512\times2+256\times3+32\times2=1\,856$ bit。

所提协议中，在注册阶段，主验证设备向服务器发送消息$\langle \mathrm{ID_C},\mathrm{pk}'_\mathrm{B}\rangle$并保存，其中，$\mathrm{pk}'_\mathrm{B}$为SM2公钥，长度为512 bit，共544 bit。在恢复阶段，备份验证设备与密钥服务器之间的通信消息为$\langle \mathrm{ID},a,b\rangle$，其中，$a$和$b$为OPRF中的群元素，长度为256 bit，服务器与备份验证设备之间的通信消息为$\langle \tilde{C},\tilde{k},\mathrm{ch},\sigma\rangle$，其中，$\tilde{C}$为BKEM的封装盲化值，长度为1 024 bit，$\tilde{k}$为BKEM的解封装盲化值，长度为1 024 bit，ch是长度为256 bit的挑战，σ是SM2签名，长度为512 bit，共$1\,024\times2+512+256\times3+32=3\,360$ bit。

在通信开销方面，恢复阶段中所提协议较之文

献[11]略显不足，但在注册阶段具有一定优势。除此之外，所提协议在服务器端降低了存储开销，且可保证所有认证器丢失或被捕时仍能提供安全防护。

6.3　实际应用场景部署设计

本节以智能电网工控系统为例，讨论账户恢复方案的实际部署。

1) 场景描述：在某省级智能电网工控系统，涵盖上千个变电站终端设备，运维人员通过手持验证设备（PAD/BAD）远程登录系统进行设备监控与维护。其核心需求如下。

① 高安全性：符合国密标准，抵御伪造身份攻击（如恶意运维人员冒用账户）。

② 快速恢复：设备丢失后，迅速完成账户恢复，避免影响故障抢修。

③ 低部署成本：兼容现有变电站终端。协议基于SM2国密算法，需硬件支持SM2-P-256椭圆曲线。目前国内工业设备（如华为工业网关、研华工控机）已普遍集成国密加速芯片。但部分设备仅支持RSA算法，可能需要额外部署算法转换网关进行升级。

2) 部署环境：客户端可采用手持终端（50%搭载国密芯片，50%旧设备通过网关转换）设备模拟；服务器分为AAS集群与KS集群，可部署于华为云Stack。智能电网工控系统部署设计如图7所示。

3) 部署成本：从硬件设备、能源消耗与维护成本3个方面讨论部署成本。

① 硬件设备：协议基于SM2国密算法设计，需支持SM2-P256椭圆曲线。目前国内工业设备（如华为工业网关、研华工控机）已普遍集成国密加速芯片，但老旧设备可能需额外升级。若系统需兼容非国密设备（如采用ECDSA的国外设备），需部署算法转换网关，增加约15%的部署成本。但长期来看，国密标准化可降低合规性风险，抵消初期投入。

② 能源消耗：客户端Raspberry Pi 4上单次恢复操作（含OPRF、BKEM、FE）耗时为286.88 ms，在高负载平均功耗为5 W的情况下能耗约为1.4 J；服务器端i9-13900K处理单次恢复请求需12.47 ms，按10 000次/天计算，月均电费增加约50元（以0.8元/(kW·h)计算）。

③ 维护成本：协议采用模块化设计，密钥服

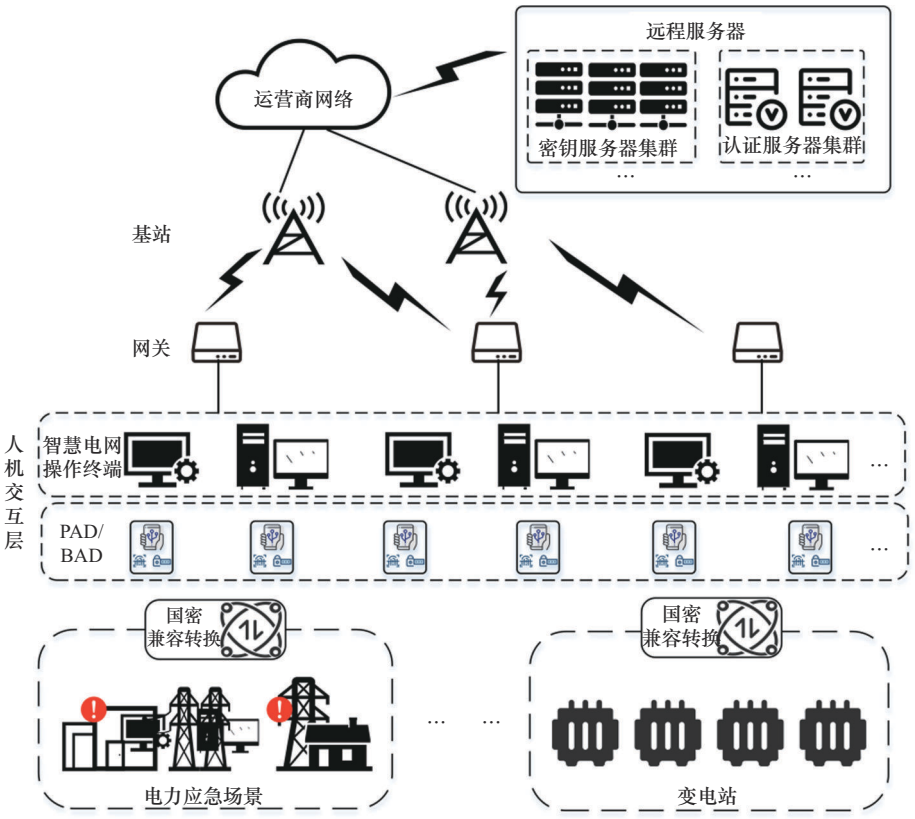

图7　智能电网工控系统部署设计

务器与认证授权服务器功能解耦，可独立部署或集成于现有工业控制系统，减少额外运维负担。此外，生物特征与口令结合的密钥恢复机制降低了人工介入频率，进一步优化长期维护成本。

6.4 可扩展性与容错性分析

1) 本文方案的可扩展性评估聚焦于协议在设备规模增长时的性能表现。

① 服务器端负载：在恢复阶段服务器可能需处理大量并发请求，其核心操作如SM2签名验证、BKEM解封装的单次开销为12.47 ms。假设服务器采用分布式架构，通过负载均衡技术可将计算压力分散至多节点，理论上支持千级并发恢复请求。

② 客户端扩展能力：客户端如备份验证设备的密钥派生与签名操作均为本地计算，无状态依赖，因此设备数量增加不会显著影响单设备性能。

③ 存储开销：注册阶段用户向服务器发送544 bit的恢复凭证，存储开销与用户数量呈线性增长但增速平缓，适合大规模工业互联网场景。

2) 本文方案的容错性主要体现在协议对设备故障及生物特征波动时的功能表现。

① 设备故障恢复：协议本身既支持验证设备丢失或故障时所采用的安全账户恢复，也可扩展为多备份验证设备注册进一步避免单点故障。

② 生物特征容错：模糊提取器通过纠错码技术容忍生物特征采样误差（如指纹识别偏差），确保生物密钥恢复成功率高于99.7%[31]，满足工业场景的可靠性要求。

综上，所提协议在保证安全性的同时，兼顾了实际部署的经济性、规模扩展的灵活性及异常场景的容错能力，为工业互联网账户恢复提供了切实可行的解决方案。

7 结束语

为应对工业互联网中身份伪造和数据泄露等账户安全问题，本文提出基于SM2 ARKG的账户恢复协议。该协议采用基于BKEM的ARKG新构造，私钥需服务器与备份设备协同生成，避免单方生成密钥的信任风险，并实现隐式双向认证。此外，结合模糊提取器将生物特征嵌入密钥派生过程，抵抗备份验证设备丢失攻击。引入不经意伪随机函数通过口令恢复种子私钥，避免静态存储，兼顾主验证设备丢失防护与密钥安全。本文对所提出的协议进行了全面的安全性分析和性能比较。结果表明，所提ARKG构造相较于现有方案具有更强的安全性，尤其是弥补了ARKG在抵御验证设备丢失攻击和认证安全性方面的不足。然而，尽管BKEM和FE模块显著提升安全性，但不可避免地增加了整体计算开销。未来工作可在安全性优势的基础上优化计算效率，并探索轻量级、后量子安全的账户恢复协议，应对未来量子带来的新威胁。

附录1 BKEM安全性证明

定义2 BKEM不可区分性。设BKEM Π(Setup,KeyGen,Encap,Blind,Decap,Unblind)是一个盲化的KEM，任何敌手$\mathcal{A}$对BKEM获得r个盲化封装及其盲化解封装元组的优势是

$$\mathrm{Adv}^{\mathrm{IND}}_{\mathrm{BKEM}}(\mathcal{A},r)=2\left|\Pr\left[\mathrm{Exp}^{\mathrm{IND}}_{\mathrm{BKEM}}(\mathcal{A},r)=1\right]-\frac{1}{2}\right| \tag{11}$$

其中，$\mathrm{Exp}^{\mathrm{IND}}_{\mathrm{BKEM}}(\mathcal{A},r)$如图8所示。

$\underline{\mathrm{Exp}^{\mathrm{IND}}_{\mathrm{BKEM}}(\mathcal{A},r)}$:
1) $b\leftarrow_{\$}\{0,1\}$
2) $(\mathrm{pk},\mathrm{sk})\leftarrow\mathrm{KeyGen}$
3) $(C,k_1)\leftarrow\mathrm{Encap}_{\mathrm{pk}}$
4) $k_0\leftarrow_{\$}\mathcal{G}$
5) for $j\in\{1,\cdots,r\}$ do
6) $(\tilde{C}_j,k'_j)\leftarrow\mathrm{Blind}_{\mathrm{pk}}(C)$
7) $\tilde{k}_j\leftarrow\mathrm{Decap}_{\mathrm{sk}}(\tilde{C}_j)$
8) $b'\leftarrow\mathcal{A}(\mathrm{pk},C,k_b,\{(\tilde{C}_j,\tilde{k}_j)\}_{1\le j\le r})$
9) return $b'\stackrel{?}{=}b$

图8 BKEM不可区分性实验

定义3 ε-盲化封装。给定BKEM封装密钥pk和原始密钥k_0的封装C_0，如果以下2个分布之间的统计距离不超过ε，则BKEM被称为具有ε-盲化封装。

分布X：$X=\{C_0+C'\|k'\leftarrow K_R,C'\leftarrow\mathrm{Enc}(\mathrm{pk},k')\}$，即用随机盲化值$k'$的封装$C'$与$C_0$结合后的结果。

分布Y：$Y=\{C|k'\leftarrow K_R,C'\leftarrow\mathrm{Enc}(\mathrm{pk},k_0+k')\}$，即对经过随机盲化的原始密钥$k+k'$进行封装的结果。

该性质确保了盲化算法的输出除了一个小概率ε之外，看起来像是封装算法的新鲜输出。

定理2 设SM2的加密方案满足$(1-\varepsilon_3)$-正确性，其中ε_3是一个可忽略的错误概率。设原始密钥k和盲化值k'分别从密钥空间K_F和K_R中均匀随机采样，BKEM具有ε_1盲化封装和ε_2盲化密钥。对于任意敌手$\mathcal{A}$，可以获取r个盲化封装及其对应的盲化解封装样本，存在一个针对SM2加密方案的IND-CCA敌手$\mathcal{B}$，使$\mathcal{A}$对BKEM的攻击成功率$\mathrm{Adv}^{\mathrm{IND}}_{\mathrm{BKEM}}(\mathcal{A},r)$可以由以下不等式进行定义

$$\mathrm{Adv}_{\mathrm{BKEM}}^{\mathrm{IND}}(\mathcal{A},r)\leqslant 2(r+1)(\varepsilon_1+\varepsilon_2+\varepsilon_3)+\mathrm{Adv}_{\mathrm{SM2PKE}}^{\mathrm{IND\text{-}CCA}}(\mathcal{B}) \tag{12}$$

这个不等式将BKEM的安全性分解为3个部分，其中，ε_1和ε_2表示盲化封装和盲化密钥的不可区分性差异，ε_3表示加密方案的正确性误差，$\mathrm{Adv}_{\mathrm{SM2PKE}}^{\mathrm{IND\text{-}CCA}}(\mathcal{B})$是敌手$\mathcal{B}$对SM2加密方案的IND-CCA攻击成功率。如果这些值都很小，则BKEM的安全性就很高。

证明

Game_0：第一个游戏是$\mathrm{Adv}_{\mathrm{BKEM}}^{\mathrm{IND}}(\mathcal{A},r)$实验。设事件$E_0$（对于每个$\mathrm{Game}_i$，$E_i$表示在该游戏中敌手成功猜测的事件）表示敌手$\mathcal{A}$的猜测$b'$与实际的$b$值相同，根据IND优势定义，有

$$\mathrm{Adv}_{\mathrm{BKEM}}^{\mathrm{IND}}(\mathcal{A},r)=2\left|\Pr[E_0]-\frac{1}{2}\right| \tag{13}$$

Game_1：该游戏与Game_0的区别在于，敌手的盲化密钥是始密钥和盲化值的和，而不是盲化封装的解密值。具体来说，设C是与原始密钥k对应的封装，对于每个$j(1\leqslant j\leqslant r)$，$C_j'$是与盲化值$k_j'$对应的封装，$C_j'+C$是盲化的封装。当敌手$\mathcal{A}$查询用户$j$的盲化密钥时，该游戏输出$k+k_j'$。根据SM2在椭圆曲线上的加法同态性，如果$C$和$C_j'$都正确解密为其对应的消息，那么在$\mathrm{Game}_1$和$\mathrm{Game}_0$中，输出的盲化密钥将是相同的。这说明在理想情况下，Game_1与Game_0之间的差异取决于解密错误的概率。因此，误差上界表示为

$$\left|\Pr[E_2]-\Pr[E_1]\right|\leqslant 1-(1-\varepsilon_3)^{r+1}\approx(r+1)\varepsilon_3 \tag{14}$$

Game_2：该游戏与Game_1的区别在于，敌手获得的盲化封装和盲化密钥不再依赖于原始密钥，而是独立且随机的。具体来说，对于每个$j(1\leqslant j\leqslant r)$：1) 当敌手$\mathcal{A}$查询用户$j$的盲化封装时，该游戏首先选择一个随机的盲化密钥$\widetilde{k}_j\leftarrow_S S$，然后使用该随机密钥$\widetilde{k}_j$生成一个新的封装$\widetilde{C}_j$发送给$\mathcal{A}$；2) 当敌手$\mathcal{A}$查询用户$j$的盲化密钥时，该游戏直接输出$\widetilde{k}_j$。

步骤1　首先证明敌手$\mathcal{A}$在Game_1和Game_2中观察到的数据分布之间的差异性。具体来说，证明在Game_1中生成的真实盲化密钥和盲化封装与在Game_2中生成修改后的盲化密钥和封装对之间的统计距离是很接近的。

设Game_1中的盲化密钥和盲化封装对统计分布$X=\{(k_0+k',C_0+C')\|k'\leftarrow K_R,C'\leftarrow\mathrm{Enc}(\mathrm{pk},k')\}$，$\mathrm{Game}_2$中修改后的盲化密钥和封装对统计分布$Y=\{(\tilde{k},\tilde{C})|\tilde{k}\leftarrow S,\tilde{C}\leftarrow\mathrm{Enc}(\mathrm{pk},\tilde{k})\}$。定义一个中间分布$Z=\{(k_0+k',C)\|k'\leftarrow K_R,C\leftarrow\mathrm{Enc}(\mathrm{pk},k_0+k')\}$。计算$X$和$Y$之间的统计距离：$\Delta(X,Y)\leqslant\Delta(X,Z)+\Delta(Z,Y)$，其中，$\Delta(X,Y)$表示$X$和中间分布$Z$之间的统计距离，根据前文分析可得其上界为$\varepsilon_1$；$\Delta(Z,Y)$表示$Z$和$Y$之间的距离，通过对各分部求和计算，并利用盲化密钥集合S的性质可得其上界为ε_2。因此有$\Delta(X,Y)\leqslant\varepsilon_1+\varepsilon_2$。对于$r$个样本，有

$$\left|\Pr[E_3]-\Pr[E_2]\right|\leqslant r(\varepsilon_1+\varepsilon_2) \tag{15}$$

步骤2　设存在一个针对SM2.PKE的IND-CCA安全性的敌手$\mathcal{B}$，使

$$2\left|\Pr[E_2]-\frac{1}{2}\right|=\mathrm{Adv}_{\mathrm{SM2PKE}}^{\mathrm{IND\text{-}CCA}}(\mathcal{B}) \tag{16}$$

首先构造一个规约$\mathcal{B}$，并通过运行$\mathcal{A}$来与IND-CCA游戏交互从而模拟A的行为，具体过程如下。

1) $\mathcal{B}$通过抛硬币决定一个随机比特b，该比特将决定在IND-CCA游戏中，攻击者将面对的挑战密文是与哪个密钥相关的。

2) $\mathcal{B}$向IND-CCA挑战者请求一个公钥，并作为封装密钥传递给$\mathcal{A}$来模拟BKEM的公钥。

3) $\mathcal{B}$随机选择2个组密钥k_0和k_1，并作为挑战查询传递给IND-CCA挑战者，从而获得一个封装密文C，然后将该密文传递给$\mathcal{A}$。

4) $\mathcal{B}$继续模拟BKEM游戏中的盲化和解封装操作，生成一个随机的盲化密钥$\tilde{k}$并使用封装算法生成一个盲化封装$\tilde{C}$，并传递给$\mathcal{A}$。除此之外，$\mathcal{B}$还需确保能够模拟解密预言机查询的响应。对于每个$\mathcal{A}$提交的密文查询，$\mathcal{B}$需要能够使用SM2的解密预言机来提供合适的解密响应。

5) 当$\mathcal{A}$请求盲化密钥时，$\mathcal{B}$将b所对应的密钥k_b发送给$\mathcal{A}$作为响应。当$\mathcal{A}$查询解密预言机时，$\mathcal{B}$使用其在IND-CCA游戏中访问的解密预言机来解密$\mathcal{A}$提交的密文，并返回对应的明文。

6) 当$\mathcal{A}$返回它的猜测b'时，$\mathcal{B}$将$1\oplus b\oplus b'$发送给挑战者。该操作的目的在于确保$\mathcal{B}$的输出与$\mathcal{A}$的猜测结果正确对应。

$\mathcal{B}$的构造确保了当$\mathcal{B}$所接收到的挑战密文是对应于真实密钥k_b时，$\mathcal{B}$完美地模拟了A在Game_2中所看到的输入。这意味着$\mathcal{A}$无法区分Game_2中的输入是否是真实的密钥还是随机的密钥。

由于SM2的PKE算法满足IND-CCA[27]安全性，因此有

$$\mathrm{Adv}_{\mathrm{BKEM}}^{\mathrm{IND}}(\mathcal{A},r)=2(r+1)(\varepsilon_1+\varepsilon_2+\varepsilon_3)+\mathrm{Adv}_{\mathrm{SM2PKE}}^{\mathrm{IND\text{-}CCA}}(\mathcal{B}) \tag{17}$$

附录2　ARKG安全性证明

定义4　ARKG公钥不可链接性（PKU）。公钥不可链接性指敌手无法区分来自某个固定分布D（即种子密钥对的分布）和其派生密钥对的分布。向敌手提供种子公钥和预言机，该预言机输出派生密钥对或从分布D进行采样的样本。该安全实验如图9所示，敌手优势定义为

$$\mathrm{Adv}_{\mathrm{SM2.ARKG}}^{\mathrm{PKU}}(\mathcal{A})=\left|\Pr[\mathrm{Exp}_{\mathrm{SM2.ARKG}}^{\mathrm{PKU}}(\mathcal{A})=1]-\frac{1}{2}\right| \tag{18}$$

定义5　ARKG私钥安全性（SKS）。私钥安全性指敌手在不知道种子私钥的情况下无法创建有效的派生密钥对。向敌手提供种子公钥和预言机，DerivePK预言机可以派生公钥，DeriveSK预言机可以生成对应的私钥，但只能在通

过DerivePK预言机生成的凭证上调用。该安全实验如图10所示。如果敌手成功生成有效的(pk,sk,cred)，则实验成功，因此敌手优势定义为

$$\mathrm{Adv}^{\mathrm{SKS}}_{\mathrm{SM2.ARKG}}(\mathcal{A}) = \left|\Pr\left[\mathrm{Exp}^{\mathrm{SKS}}_{\mathrm{SM2.ARKG}}(\mathcal{A}) = 1\right]\right| \tag{19}$$

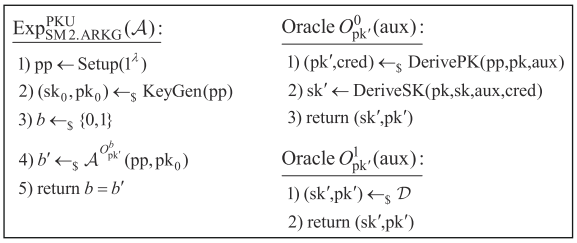

图9 公钥不可链接性实验

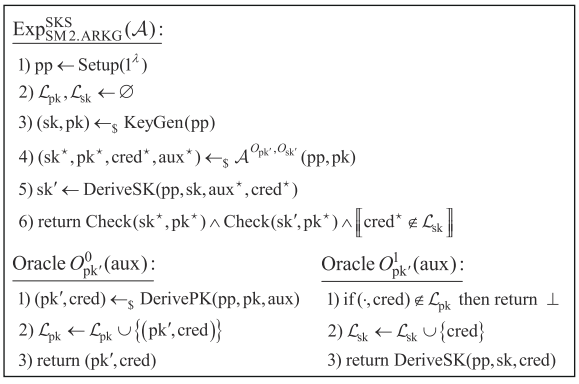

图10 私钥安全性实验

定理3 对于任意敌手$\mathcal{A}$，可以向预言机$O^{0}_{\mathrm{pk}'}$进行最多q次查询，存在有效算法$\mathcal{B}_0,\mathcal{B}_1,\mathcal{B}_2$使

$$\mathrm{Adv}^{\mathrm{PKU}}_{\mathrm{SM2.ARKG}}(\mathcal{A}) \leqslant q \cdot (\mathrm{Adv}^{\mathrm{IND}}_{\mathrm{BKEM}}(\mathcal{B}_0) + \mathrm{Adv}_{\mathrm{PRF}}(\mathcal{B}_1) + \mathrm{Adv}_{\mathrm{FE}}(\mathcal{B}_2)) \tag{20}$$

证明

Game_0：第一个游戏是$\mathrm{Exp}^{\mathrm{PKU}}_{\mathrm{SM2.ARKG}}(\mathcal{A})$实验。对于每个$\mathrm{Game}_i$，$E_i$表示在该游戏中敌手成功猜测的事件，根据PKU优势定义，有

$$\mathrm{Adv}^{\mathrm{PKU}}_{\mathrm{SM2.ARKG}}(\mathcal{A}) = 2\left|\Pr[E_0] - \frac{1}{2}\right| \tag{21}$$

Game_1：该游戏与Game_0的区别在于，BKEM.Encaps生成的密钥k在调用$O^{0}_{\mathrm{pk}'}$时被替换为一个真正的随机值。此时，Game_1与Game_0之间的差异取决于BKEM的IND安全性。因此，对于一个有效的IND敌手$\mathcal{B}_0$可以进行最多q次查询，有

$$\left|\Pr[E_1] - \Pr[E_0]\right| \leqslant q \cdot \mathrm{Adv}^{\mathrm{IND}}_{\mathrm{BKEM}}(\mathcal{B}_0) \tag{22}$$

Game_2：该游戏与Game_1的区别在于，模糊提取器FE生成的生物特征密钥R被替换为等长的伪随机值。此时，Game_2与Game_1之间的差异取决于FE的安全性。因此，对于一个有效的多项式敌手$\mathcal{B}_1$，可以进行最多q次查询，有

$$\left|\Pr[E_2] - \Pr[E_1]\right| \leqslant q \cdot \mathrm{Adv}_{\mathrm{FE}}(\mathcal{B}_1) \tag{23}$$

Game_3：该游戏与Game_2的区别在于，所有通过伪随机函数PRF生成的τ值都被替换为真正的随机值。此时，Game_3与Game_2之间的差异取决于PRF的伪随机性。因此，对于一个有效的多项式敌手$\mathcal{B}_2$，可以进行最多q次查询，有

$$\left|\Pr[E_3] - \Pr[E_2]\right| \leqslant q \cdot \mathrm{Adv}_{\mathrm{PRF}}(\mathcal{B}_1) \tag{24}$$

综上，通过结合这3个游戏的约束，可以得到敌手优势为

$$\mathrm{Adv}^{\mathrm{PKU}}_{\mathrm{SM2.ARKG}}(\mathcal{A}) \leqslant q \cdot (\mathrm{Adv}^{\mathrm{IND}}_{\mathrm{BKEM}}(\mathcal{B}_0) + \mathrm{Adv}_{\mathrm{PRF}}(\mathcal{B}_1) + \mathrm{Adv}_{\mathrm{FE}}(\mathcal{B}_2)) \tag{25}$$

定理4 对于任意敌手$\mathcal{A}$，可以向预言机$O^{0}_{\mathrm{pk}'}$进行最多q次查询，存在有效算法$\mathcal{B}_0,\mathcal{B}_1,\mathcal{B}_2,\mathcal{B}_3$使

$$\mathrm{Adv}^{\mathrm{SKS}}_{\mathrm{SM2.ARKG}}(\mathcal{A}) \leqslant q \cdot (\mathrm{Adv}^{\mathrm{IND}}_{\mathrm{BKEM}}(\mathcal{B}_0) + \mathrm{Adv}_{\mathrm{FE}}(\mathcal{B}_1) + \mathrm{Adv}_{\mathrm{PRF}}(\mathcal{B}_2) + \mathrm{Adv}_{\mathrm{ECDLP}}(\mathcal{B}_3)) + \frac{1}{2^n} \tag{26}$$

证明

Game_0：第一个游戏是$\mathrm{Exp}^{\mathrm{SKS}}_{\mathrm{SM2.ARKG}}(\mathcal{A})$实验。对于每个$\mathrm{Game}_i$，$E_i$表示在该游戏中敌手成功猜测的事件，根据PKU优势定义，有

$$\mathrm{Adv}^{\mathrm{SKS}}_{\mathrm{SM2.ARKG}}(\mathcal{A}) = \left|\Pr[E_0]\right| \tag{27}$$

Game_1：该游戏与Game_0的区别在于，将$O^{0}_{\mathrm{pk}'}$中BKEM.Encaps生成的密钥k、FE输出的生物密钥R以及伪随机函数PRF的输出τ都替换为等长的随机值，并将$O_{\mathrm{sk}'}$中对PRF的调用替换为对存储的τ值的查找。此时，Game_1与Game_0之间的差异取决于BKEM的IND安全性，以及FE和PRF的安全性。因此，对于有效的多项式敌手$\mathcal{B}_0,\mathcal{B}_1,\mathcal{B}_2$，可以进行最多$q$次查询，有

$$\left|\Pr[E_1] - \Pr[E_0]\right| \leqslant q \cdot (\mathrm{Adv}^{\mathrm{IND}}_{\mathrm{BKEM}}(\mathcal{B}_0) + \mathrm{Adv}_{\mathrm{FE}}(\mathcal{B}_1) + \mathrm{Adv}_{\mathrm{PRF}}(\mathcal{B}_2)) \tag{28}$$

Game_2：该游戏与Game_1的区别在于，将$O_{\mathrm{pk}'}$中的$\mathrm{pk}' \leftarrow [\tau]G + \mathrm{pk}$替换为KeyGen，即直接生成新的密钥对，存储生成的私钥sk'、cred和τ。将$O_{\mathrm{sk}'}$中的$\mathrm{sk}' \leftarrow \tau + \mathrm{sk}$替换为对存储的$\mathrm{sk}'$的查找。在SM2中，由于$\tau$是一个独立的随机数，因此$\mathrm{sk}'$的分布实际上是一个均匀分布在私钥空间上的随机变量。这与直接从私钥空间中随机选择一个新的私钥（完全新生成的密钥对）在统计上是不可区分的。因此，有

$$\left|\Pr[E_2] - \Pr[E_1]\right| \leqslant \frac{1}{2^n} \tag{29}$$

其中，n是椭圆曲线的阶。

Game_3：该游戏模拟执行椭圆曲线离散对数问题（ECDLP, elliptic curve discrete logarithm problem）。假设已知种子公钥pk，若敌手能够通过预言查询得到sk，则认为敌手能够解决ECDLP。对于有效的多项式敌手$\mathcal{B}_3$，可以进行最多q次查询，有

$$\left|\Pr[E_3] - \Pr[E_2]\right| \leqslant q \cdot \mathrm{Adv}_{\mathrm{ECDLP}}(\mathcal{B}_3) \tag{30}$$

综上，通过结合以上游戏的约束，可以得到敌手优势为

$$\mathrm{Adv}^{\mathrm{SKS}}_{\mathrm{SM2.ARKG}}(\mathcal{A}) \leqslant q \cdot (\mathrm{Adv}^{\mathrm{IND}}_{\mathrm{BKEM}}(\mathcal{B}_0) + \mathrm{Adv}_{\mathrm{FE}}(\mathcal{B}_1) + \mathrm{Adv}_{\mathrm{PRF}}(\mathcal{B}_2) + \mathrm{Adv}_{\mathrm{ECDLP}}(\mathcal{B}_3)) + \frac{1}{2^n} \tag{31}$$

参考文献：

[1] 刘奇旭, 肖聚鑫, 谭耀康, 等. 工业互联网流量分析技术综述[J]. 通信学报, 2024, 45(8): 221-237.

LIU Q X, XIAO J X, TAN Y K, et al. Survey of industrial Internet traffic analysis technology[J]. Journal on Communications, 2024, 45(8): 221-237.

[2] 杨婷, 张嘉元, 黄在起, 等. 工业控制系统安全综述[J]. 计算机研究与发展, 2022, 59(5): 1035-1053.
YANG T, ZHANG J Y, HUANG Z Q, et al. Survey of industrial control systems security[J]. Journal of Computer Research and Development, 2022, 59(5): 1035-1053.

[3] 姜奇, 杨雪, 王金花, 等. 面向车联网的抗设备捕获认证密钥协商协议[J]. 中国科学: 信息科学, 2022, 52(12): 2351-2370.
JIANG Q, YANG X, WANG J H, et al. Device capture resilient authentication and key agreement protocol for IoV[J]. Scientia Sinica (Informationis), 2022, 52(12): 2351-2370.

[4] 姜奇, 文悦, 张瑞杰, 等. 面向智能手机的自适应触屏持续认证方案[J]. 电子学报, 2022, 50(5): 1131-1139.
JIANG Q, WEN Y, ZHANG R J, et al. An adaptive touchscreen based continuous authentication scheme for smart phones[J]. Acta Electronica Sinica, 2022, 50(5): 1131-1139.

[5] NOSOUHI M R, BAIG Z, DOSS R, et al. Towards availability of strong authentication in remote and disruption-prone operational technology environments[C]//Proceedings of the 19th International Conference on Availability, Reliability and Security. New York: ACM Press, 2024: 1-11.

[6] CIRANI S, PICONE M, GONIZZI P, et al. IoT-OAS: an OAuth-based authorization service architecture for secure services in IoT scenarios[J]. IEEE Sensors Journal, 2015, 15(2): 1224-1234.

[7] SCIANCALEPORE S, PIRO G, CALDAROLA D, et al. OAuth-IoT: an access control framework for the Internet of things based on open standards[C]//Proceedings of the 2017 IEEE Symposium on Computers and Communications (ISCC). Piscataway: IEEE Press, 2017: 676-681.

[8] HARDT D. The OAuth 2.0 authorization framework[R]. 2012.

[9] LYASTANI S G, SCHILLING M, NEUMAYR M, et al. Is FIDO2 the kingslayer of user authentication? A comparative usability study of FIDO2 passwordless authentication[C]//Proceedings of the 2020 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2020: 268-285.

[10] LUNDBERG E, NILSSON D. Webauthn recovery extension[R]. 2019.

[11] FRYMANN N, GARDHAM D, KIEFER F, et al. Asynchronous remote key generation: an analysis of yubico's proposal for W3C WebAuthn[C]//Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2020: 939-954.

[12] FRYMANN N, GARDHAM D, MANULIS M, et al. Generalised asynchronous remote key generation for pairing-based cryptosystems[C]//Applied Cryptography and Network Security. Berlin: Springer, 2023: 394-421.

[13] FRYMANN N, GARDHAM D, MANULIS M. Asynchronous remote key generation for post-quantum cryptosystems from lattices[C]//Proceedings of the 2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P). Piscataway: IEEE Press, 2023: 928-941.

[14] BRENDEL J, CLERMONT S, FISCHLIN M. Post-quantum asynchronous remote key generation for FIDO2[C]//Advances in Cryptology – ASIACRYPT 2024. Berlin: Springer, 2024: 465-493.

[15] STEBILA D, WILSON S. Quantum-safe account recovery for WebAuthn[C]//Proceedings of the 19th ACM Asia Conference on Computer and Communications Security. New York: ACM Press, 2024: 1814-1830.

[16] 国家密码管理局. GB/T 32918.2-2016 信息安全技术 SM2 椭圆曲线公钥密码算法[S]. 北京: 中国标准出版社, 2016.
State Cryptography Administration. GB/T 32918.2-2016 Information security technology SM2 elliptic curve public key cryptography algorithm[S]. Beijing: Standards Press of China, 2016.

[17] BOYD C, DAVIES G T, GJØSTEEN K, et al. Offline assisted group key exchange[C]//Developments in Language Theory. Berlin: Springer, 2018: 268-285.

[18] DODIS Y, OSTROVSKY R, REYZIN L, et al. Fuzzy extractors: how to generate strong keys from biometrics and other noisy data[J]. SIAM Journal on Computing, 2008, 38(1): 97-139.

[19] FREEDMAN M J, ISHAI Y, PINKAS B, et al. Keyword search and oblivious pseudorandom functions[C]//Theory of Cryptography. Berlin: Springer, 2005: 303-324.

[20] CAHILL C P, HUGHES J, LOCKHART H, et al. Assertions and protocols for the oasis security assertion markup language (SAML) V2.0[S]. Burlington: Organization for the Advancement of Structured Information Standards (OASIS), 2005.

[21] BALFANZ D, CZESKIS A, HODGES J, et al. Web authentication: an API for accessing public key credentials level 1[R]. 2019.

[22] FRYMANN N, GARDHAM D, MANULIS M. Unlinkable delegation of WebAuthn credentials[C]//European Symposium on Research in Computer Security. Berlin: Springer, 2022: 125-144.

[23] BONEH D, LYNN B, SHACHAM H. Short signatures from the Weil pairing[C]//Advances in Cryptology — ASIACRYPT 2001. Berlin: Springer, 2001: 514-532.

[24] WATERS B. Efficient identity-based encryption without random oracles[C]//Advances in Cryptology – EUROCRYPT 2005. Berlin: Springer, 2005: 114-127.

[25] CAMENISCH J, LYSYANSKAYA A. Signature schemes and anonymous credentials from bilinear maps[C]//Advances in Cryptology – CRYPTO 2004. Berlin: Springer, 2004: 56-72.

[26] HANSER C, SLAMANIG D. Structure-preserving signatures on equivalence classes and their application to anonymous credentials[C]//Advances in Cryptology – ASIACRYPT 2014. Berlin: Springer, 2014: 491-511.

[27] POINTCHEVAL D, SANDERS O. Short randomizable signatures[C]//Topics in Cryptology - CT-RSA 2016. Berlin: Springer, 2016: 111-126.

[28] WILLIAMS T J. The Purdue enterprise reference architecture[J]. Computers in Industry, 1994, 24(2-3): 141-158.

[29] ZHANG Z F, YANG K, ZHANG J, et al. Security of the SM2 signature scheme against generalized key substitution attacks[C]//Security Standardisation Research. Berlin: Springer, 2015: 140-153.

[30] JARECKI S, KIAYIAS A, KRAWCZYK H, et al. Highly-efficient and composable password-protected secret sharing (or: how to protect your Bitcoin wallet online) [C]//Proceedings of the 2016 IEEE European Symposium on Security and Privacy (EuroS&P). Piscataway: IEEE Press, 2016: 276-291.

[31] LIBERATI E, VISINTIN A, LAZZERETTI R, et al. X-lock: a secure XOR-based fuzzy extractor for resource constrained devices[C]//Applied Cryptography and Network Security. Berlin: Springer, 2024: 183-210.

[32] KRAWCZYK H, ERONEN P. HMAC-based extract-and-expand key derivation function (HKDF)[R]. 2010.

[33] HANSEN T. US secure hash algorithms (SHA and HMAC-SHA) [R]. 2006.

[作者简介]

肖浩（1989– ），男，湖南益阳人，西安电子科技大学博士生，主要研究方向为工业互联网、密码协议等。

余增文（1985– ），男，陕西安康人，西安电子科技大学博士生，北京计算机技术及应用研究所高级工程师，主要研究方向为大数据、数据治理、数据安全等。

杨雪（1999– ），女，陕西渭南人，西安电子科技大学博士生，主要研究方向为多因子认证协议、后量子密码学等。

李兴华（1978– ），男，河南南阳人，博士，西安电子科技大学教授、博士生导师，主要研究方向为无线网络安全、隐私保护、数据安全。

姜奇（1983– ），男，安徽滁州人，博士，西安电子科技大学教授、博士生导师，主要研究方向为密码协议、物联网安全等。

马建峰（1963– ），男，陕西西安人，博士，西安电子科技大学教授、博士生导师，主要研究方向为应用密码学、无线网络安全等。