群智感知中基于区块链的带时效签密方案

王利朋[1), 2)] 陈 钟[1)] 关 志[1)] 李青山[1)]

[1)] (北京大学信息科学技术学院 北京 100871)

[2)] (郑州师范学院信息科学与技术学院 郑州 450044)

摘 要 用户利用手机或者智能手环等终端设备收集环境数据，但数据在传输过程中极容易遭受窃听、篡改等威胁. 基于椭圆曲线提出一种无证书签密方案，以保障信息的安全性和可验证性. 基于离散对数和计算性 Diffie-Hellman 问题，在随机预言机模型下证明了方案的机密性和不可伪造性，此外新方案具有公开验证性和匿名性等安全属性. 为了方便对终端设备的精确控制，提出一种适配于签密方案的节点退出机制，在该机制中，基于区块链进行公钥时效管理，确保设备按照配置策略退出. 公钥信息存储在区块链中，避免了针对公钥信息恶意篡改的问题. 由智能合约更新公钥有效性，无需人工参与，确保时效管理模块的可信性. 区块链执行公钥时效更新操作，不占用物联网设备的计算资源. 性能分析显示，新方案具有较短的密钥长度，较低的计算复杂度. 在实验仿真部分，首先给出了签密算法各个步骤执行时间的对比结果，并分析了数据量对签密算法性能的影响. 然后给出了引入时效管理模块后签密算法执行的时间，结果显示签密步骤性能损失约为 7%，解签密步骤性能损失不到 1%，而且两个步骤执行时间均不超过 120 ms，能够有效适配到物联网应用场景中.

关键词 群智感知；区块链；时效性；离散对数；签密

中图法分类号 TP393.08 **DOI 号** 10.11897/SP.J.1016.2021.02216

Blockchain-Based Signcryption Scheme with Aging Mechanism in Crowdsensing Applications

WANG Li-Peng[1), 2)] CHEN Zhong[1)] GUAN Zhi[1)] LI Qing-Shan[1)]

[1)] (*School of Electronics Engineering and Computer Science, Peking University, Beijing* 100871)

[2)] (*College of Information Science and Technology, Zhengzhou Normal University, Zhengzhou* 450044)

Abstract Users can utilize terminal devices such as mobile phones or smart bracelets to collect surrounding data, but those data are vulnerable to network threats such as eavesdropping and tampering during data transmission. In order to guarantee the security and authenticity of the users' data, a certificate-less signcryption scheme based on the elliptic curve is proposed. The proposed scheme includes seven steps which are setup, genPartialKey, genPrivateKey, genPublicKey, signCrypt, unSignCrypt and verifySign. For the setup step, with the input of security parameters, key generation centers (KGCs) output the system master key and public parameters. During the genPartialKey phase, KGCs and users take the system master key, user identities and other parameters as inputs to generate partial keys for users. The next step is to generate private keys. The fourth step is to generate those users' public keys. For the signCrypt phase, a sender calculates the ciphertext for the original plaintext with public parameters and other information as inputs. The sixth step is to perform the decryption operation, and after that the receiver outputs the plaintext corresponding to the given ciphertext. The final step is to verify the decrypted

收稿日期：2019-12-24；在线发布日期：2020-02-07. 本课题得到国家重点研发计划(2020YFB1005404, 2018YFB0803601)、河南省高等学校重点科研项目计划(22A520048, 20B520040)资助. **王利朋**，博士研究生，主要研究领域为密码学、区块链. E-mail: wlpscu@126.com. **陈 钟**（**通信作者**），博士，教授，主要研究领域为密码学、区块链. E-mail: zhongchen@pku.edu.cn. **关志**，博士，副教授，主要研究领域为密码学. E-mail: guanzhi@pku.edu.cn. **李青山**，博士研究生，主要研究领域为区块链. E-mail: liqs@infosec.pku.edu.cn.

plaintext with public keys and other parameters. All the above steps do not include bilinear pairing operations, which are time consuming. Based on the intractability of the elliptic curve discrete logarithm problem and the elliptic curve Diffie-Hellman problem, confidentiality and unforgeability of the proposed method are proved in the random oracle model. The new scheme also owns other security attributes such as public verification, anonymity, which are also discussed in the paper. For precise control of sensing devices, we propose a node withdrawal method, which can be adapted to the new signcryption scheme. The new node withdrawal method introduces a public key aging mechanism based on blockchain to guarantee that a device can exit according to system configurations. Public keys of those devices are stored in blockchain with the help of smart contracts. Because blockchain has the characteristic of non-tamperability, public keys cannot be tampered with. To remove a specific device from the system, we can set up an aging period for the public key of the device. After we store the public key along with its aging period in blockchain through a smart contract, that information for the device can be deleted automatically without human participation to ensure the credibility of the process. Without the public key in blockchain, the device cannot perform the signCrypt operation and the unSignCrypt operation. What is more, the aging operation is performed by blockchain without power consumption of IoT devices. Performance analysis shows that the proposed signcryption scheme with a shorter key length has lower computational complexity. In the simulation part, comparison results for execution time are firstly given, and then performance impacts of data volume are also analyzed. With introduction of the aging mechanism, performance of the signCrypt step in the proposed scheme is decreased by about 7%, and one of the unSignCrypt steps is decreased by less than 1%. Even so, each of the above two steps consumes less than 120ms, which can still be adapted to the IoT scenarios.

Keywords　crowdsensing; blockchain; aging; discrete logarithm; signcryption

1　引　言

在信息时代，数据安全问题日益受到人们的重视，消息进行传输时候，为了防止恶意第三方发起的对消息内容的攻击，数据安全性和可验证性变得愈加重要，而为了实现上述要求，加(解)密和数字签名是对应的两种数据保护技术。传统的消息安全传输方案中，一般先对消息进行数字签名，然后再对消息内容进行加密，当接收方收到密文信息后，再进行相反的操作，以确保消息的可信传输。然而上述方案中将加(解)密操作和签名操作进行了分离，本质上是一种对消息内容二次加工的策略，计算效率较低，因此在 1997 年，文献[1]首次提出了签密概念，能够将签名和加(解)密这两种独立操作在一个逻辑步骤中同时完成，极大地提升了算法的执行效率。

物联网技术出现的目的就是为了创建一个万物互联的世界，用户利用手机或者智能手环等终端设备，基于内置传感器作为感知设备，收集周围环境数据，并通过通信设施进行信息传输，以供数据收集者从海量数据中挖掘有用信息[2]。文献[3]提出了一种适用于物联网群智感知场景中的区块链系统，该系统基于区块链去中心化的特性，实现了分布式感知数据的记录和存储，可应用于城市噪声感知、光源自动控制等领域，该系统的框架模型如图 1 所示。该方案的步骤如下：(1) 服务器发布带奖励的感知任务；(2) 用户上传感知数据；(3) 用户组成员校验数据质量，并将校验后的数据以交易的形式发送至区块链；(4) 矿工校验交易信息；(5) 区块链分配奖励。在上述方案中，用户按照一定原则进行了分组，形成用户组，组成员之间是彼此信任的，并承担感知数据质量校验工作。这样的目的一方面能够增加系统的执行效率，另一方面能够将用户隐私数据隔离在组内，其他组成员无法获知这些数据，减少泄露的风险。用户组将感知数据校验成功后，上传至区块链，然后区块链节点从第 4 步开始执行后续操作。

在上传感知数据到区块链时，由于区块数据的公开性，用户上传的敏感数据需进行签密处理，保证数据的机密性和完整性，然后矿工可以对未加密

的非敏感数据或者元数据信息进行质量校验。文献[3]提出了一种基于双线性对的签密算法，可对敏感数据进行签密处理。由于双线性对操作比较耗时，为了提升效率，该签密算法假设组用户之间是信任的，并通过缩减参与人员数量提升算法的执行效率。需要说明的是，上述签密算法是一种无证书公钥算法，并不需要证书管理中心来管理用户公钥等信息。如果用户组中的成员节点存在恶意行为，上述算法并没有实现一种节点可信退出机制，而如果要实现这样的功能，存在两种可行的方法：

第一种方法是引入可信第三方中心节点来管理用户公钥等信息，一旦检测到恶意节点，由中心节点主动发起删除恶意节点的动作。然而对于这种实现方案，中心节点是系统的安全瓶颈所在，一旦被攻破，会对整个系统的安全造成威胁。

第二种实现方法是让用户组成员节点相互之间进行协作，一旦检测到某一节点不可信，成员节点对该节点进行广播，并删除存储在本地的该不可信节点的公钥等信息，从而达到排除不可信节点的目的。然而该方法实现复杂度较高，受限于感知设备计算能力，在物联网场景中适配范围有限。

文献[3]提出的方案主要存在以下问题：

(1) 实际应用场景中，感知设备类别多种多样，计算能力参差不齐，基于双线性对实现的签密算法，执行效率较低。

(2) 该方案假设组用户之间是可信的，用户组成员长期固定，而在实际场景中，受限于感知设备计算能力，其安全防护措施较弱，用户组成员之间的信任关系较为脆弱。一旦成员节点被攻破，缺乏一种轻量级的可信退出机制，以删除该成员节点。

针对上述问题，如果能够设计出一种轻量而且具有时效性的签密方案，一方面实现传输数据的机密性和完整性，另一方面使得感知设备按照配置策略适时退出用户组，并确保用户公钥信息真实可信，是一种可行的研究方向。

在利用区块链技术实现用户公钥时效管理功能时，可以采用下述方案：首先将用户公钥等相关信息存储到区块链中，如果需要剔除用户组中某一成员节点，只需要删除该节点存储在区块链上对应的公钥等信息即可，这是由于其它节点在执行签密和解签密操作时需要用户公钥等信息。需要说明的是，这里在区块链上执行删除操作是通过发起新的交易实现，原有历史数据并不会被真正删除。由于区块链数据的公开性，其它成员节点可以通过访问区块链，以获知公钥失效的信息，这样首先可以避免因引入第三方中心节点而带来的安全瓶颈问题，其次基于区块链技术可以保证时效管理模块的可信性。

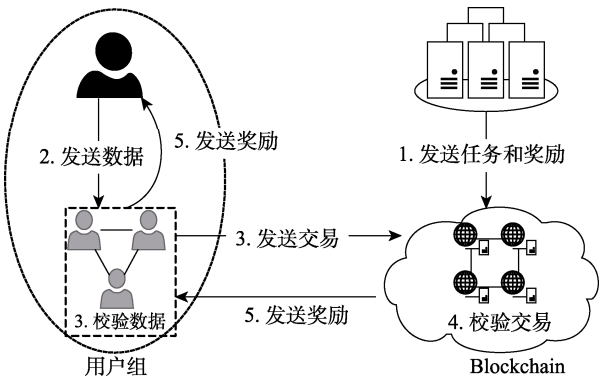

图 1 基于区块链的群智感知应用的框架模型

本文的主要贡献如下：

(1) 基于椭圆曲线提出了一种无证书签密方案，并在随机预言机模型下，基于椭圆曲线 DL 困难问题和基于椭圆曲线 CDH 困难问题，对方案的机密性和不可伪造性进行了证明，此外本文方案具有公开验证性、不可否认性、匿名性等安全属性。本文方案具有较低的计算复杂度、较短的密钥长度，能够有效适配到物联网这种计算资源稀缺的应用场景中。

(2) 将区块链技术应用到上述签密方案中，提出了一种公钥时效管理方案，实现成员节点的可信退出功能。首先将带时效性的公钥以智能合约形式存储到区块链中，基于区块链确保公钥不可篡改性，然后智能合约按照时间约定自动更新公钥有效性。更新过程无需人工参与，由区块链自动执行，保证更新过程的可信性。

(3) 仿真实验结果表明，本文提出的签密方案所需的存储空间较少，且具有较高的执行效率。引入时效管理模块后，签密步骤的性能损失不超过 8%，解签密步骤的性能损失不超过 1%。

2 相关工作

2.1 基于无证书公钥系统签密方案

文献[4]首次提出了无证书公钥系统（Certificate-less Public Key Cryptosystem, CL-PKC）的概念，在 CL-PKC 中，用户需要通过密钥生成中心（Key Generation Center, KGC）生成自己的密钥信息. CL-PKC 克服了公钥证书管理的问题，解决了因密钥托管问题而引入的安全问题，同时提高了系统的执行效率. 文献[5]首次将 CL-PKC 引入到签密领域，并提出了无证书的签密概念. 当前无证书的签

密方案，主要有基于离散对数签密方案、基于椭圆曲线签密方案以及基于双线性对签密方案，近些年来，相关研究人员又提出了具有抗量子攻击特性的签密方案. 一般来讲，上述方案对应的安全性依次提升，然而其对应的入门难度也依次增加.

近年来相关研究人员基于双线性对操作提出了多种签密方案. 文献[5]提出了一种无证书签密方案，该方案基于双线性对进行实现，具有前向安全性. 文献[6]提出了一种混合签密方法，使其能够适用于无证书的签密场景中. 文献[7]基于离散对数（Discrete Logarithm, DL）问题和 Computational Diffie-Hellman（CDH）问题提出了一种签密方案，并首次引入了公开验证性的概念，具有较高的安全性. 为了解决用户敏感数据在传输时被篡改的风险，实现数据的安全性和完整性，文献[2]提出了一种基于双线性对的签密方案，并对算法的性能进行了优化. 基于双线性对的签密方案，普遍存在计算复杂度较高的问题，在适配到对性能要求较高的场景中时，需要对其进行优化.

基于离散对数实现的签密方案，相比较基于双线性对实现的签密方案，其执行效率要高. 文献[8]基于随机预言机模型，为文献[9]构造了针对机密性和不可伪造性的攻击算法，并基于离散对数问题提出了一种改进方案，同时给出了机密性和不可伪造性证明. 文献[10]基于离散对数提出了一种能够抵制多个签名者联合攻击的多重签密方案，并在随机预言机模型下给出了不可伪造性证明. 基于离散对数的签密方案，在达到相同安全等级要求下，普遍要比基于双线性对实现方案的密钥长度要长，因此在使用此类方案时，需要重点关注该问题.

2.2　抗量子攻击的签密方案

在后量子时代，公认的主要有四种密码体制，分别是基于哈希的密码体制、基于编码的密码体制、基于格的密码体制和基于多变量的密码体制. 相关研究人员基于这些抗量子攻击的密码体制，提出了相应的签密方案，具有更高的安全性.

文献[11]提出了一种基于编码密码体制的签密方案，其依赖的数学问题是码字的译码问题，该方案具有机密性和不可伪造性，能够抵抗量子攻击，且具有较高的计算效率. 基于编码的密码体制，普遍存在密钥尺寸较长的问题，后面相关人员对其进行了改进，分别提出了 Quasi-Cyclic Medium-Density Parity-Check codes （QC-MDPC）[12]、Low Density Parity Check codes（LDPC）[13]以及 Quasi-Cyclic Low-Density Parity-Check codes （QC-LDPC）[14]等方案.

基于格的密码体制，其安全性建立在小整数问题、随机情况带错学习问题等困难问题之上，文献[15]提出了一种模糊身份格基签密方案，该方案实现了对模糊身份的签名和加密功能. 文献[16]基于 MP 陷门产生算法和盆景树格基代理方案，构造了一种基于身份的签密方案，不需要借助传统的公钥基础设施来维护数字证书. 文献[17]提出了一种标准模型下的签密方案，构建了一种基于身份的可证安全的签密方案. 当前基于格的签密方案的可证安全性主要是基于随机预言机模型，缺少标准模型下的安全性证明，且存在效率低的问题，仍具有较大的改进空间.

基于多变量密码体制的签密算法，核心映射的构造是其中的研究难点所在，文献[18]基于 LRPC（Low Rank Parity Check）码和 Cubic Simple Matrix 加密算法，结合一种具有较小密钥量的秩矩阵码，实现了一种多变量签密方案. 目前在多变量密码核心映射方面，文献[19-21]均提出了相应的改进方案，采用加方法、减方法、醋变量方法及内部扰动等方法，极大地提高了安全性.

上述抗量子密码方案虽然理论上具有较高的安全性，但文献[22]认为在可预见的未来，通用量子计算机尚不能够被制造出来，且大部分抗量子攻击的密码方案缺乏系统性证明，形成完备的理论体系仍有较长的路要走.

2.3　基于区块链的用户时效管理方案

在现实场景中，如何确保用户时效管理模块的可信性，保证数据不被恶意篡改，是需要重点解决的问题. 如果基于第三方中心节点对时效性进行管理，该中心节点容易成为系统的安全瓶颈. 例如在冷链食品管理中，第三方可以攻击中心服务器，修改食品的过期时间，如何使客户信任其时效性仍是一个重要的研究问题.

区块链是一种分布式数据库技术，本身具有去中心化、伪匿名性、可信溯源、不可篡改等特性，能够实现用户间的信任管理，目前已广泛应用于供应链管理、金融监管、电子认证、投票选举、边缘计算等领域[23-27]. 区块链主要分为公有链、联盟链和私有链[28]，当前也出现了一些基于区块链对用户公钥时效进行管理的方案[28]，区块链系统能够有效防止信息被恶意篡改，进而保证时效管理过程的可信性。

3 预备知识

3.1 困难性问题

DL 问题：设定 p 是大素数，F_p 是阶数为 p 的循环群，G 为 F_p 中的一个生成元. 给定元素组 $\langle G, aG\rangle$，求解 a，其中 $a \in Z_p^*$. 若存在算法 Γ，能够在多项式时间内解决 DL 问题，其对应的概率为 $Adv^{DL}(\Gamma) = Pr[\Gamma(G, aG) = a]$.

定义 1. DL 假设：算法 Γ 在多项式时间内解决 DL 问题的概率 $Adv^{DL}(\Gamma)$ 是可忽略的.

CDH 问题：设定 p 是大素数，F_p 是阶数为 p 的循环群，G 为 F_p 中的一个生成元. 给定元素组 $\langle G, aG, bG\rangle$，求解 abG，其中 $a, b \in Z_p^*$. 若存在算法 Γ，能够在多项式时间内解决 CDH 问题，其对应的概率为 $Adv^{CDH}(\Gamma) = Pr[\Gamma(G, aG, bG) = abG]$.

定义 2. CDH 假设：算法 Γ 在多项式时间内解决 CDH 问题的概率 $Adv^{CDH}(\Gamma)$ 是可忽略的.

3.2 系统模型

在系统实现中，参与者主要包括了签密者 a、接收者 b 和密钥生成中心 KGC，主要包括了七个步骤. 首先 KGC 执行注册（Setup）步骤，输入安全参数，输出公开参数信息和系统主密钥信息. 然后是部分密钥生成（GenPartialKey）步骤，输入用户相关信息以及系统主密钥信息，用户和 KGC 协作生成用户部分密钥信息. 第三步执行用户私钥生成（GenPrivateKey）步骤，以输出用户私钥信息. 第四步执行用户公钥生成（GenPublicKey）步骤，输出用户公钥信息. 第五步是执行签密（SignCrypt）操作，签密者 a 根据公开参数、明文等其它信息，输出密文信息. 第六步是执行解密（UnSignCrypt）操作，接收者 b 根据公开参数等信息，输出明文信息. 最后一步是执行签名校验（VerifySign）步骤，接收者 b 根据公开参数等信息对解密出的明文消息进行校验.

用户带时效性的公钥等信息以智能合约形式存储到区块链中，基于区块链不可篡改的特性，以确保用户公钥信息真实可信. 智能合约按照约定时间自动更新公钥有效性，更新过程无需人工参与，确保时效管理模块的可信性，系统模型图如图 2 所示.

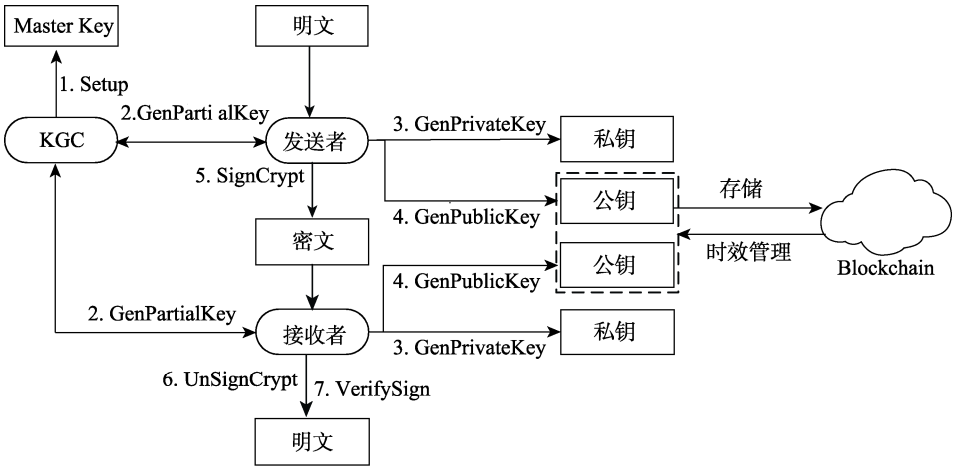

图 2 系统模型图

3.3 安全模型

按照文献[4]定义的安全模型，签密系统主要面临两种类型的攻击方式，第一种是攻击敌手能够利用其他用户的公钥攻击系统的安全性，且敌手能够替换用户的公钥信息，这种攻击方式标记为 λ_{I}. 第二种是攻击敌手能够获取系统的主密钥信息，并攻击系统的安全性，这种攻击方式标记为 λ_{II}. 签密方案应该具有适应性选择密文攻击下的机密性和适应性选择消息攻击下的不可伪造性. 下面定义本文用到的四种安全模型，详细步骤请参考文献[4].

定义 3. 攻击方式 λ_{I} 下的机密性. 敌手如果不能够通过多项式时间的计算以不可忽略优势 $Adv^{c}(\lambda)$ 赢得参考文献[4]定义的攻击方式 λ_{I} 下机密性相关游戏，则称方案具有适应性选择密文攻击下的机密性，且该攻击类型下的攻击敌手标记为 $\lambda_{\text{I-1}}$.

定义 4. 攻击方式 λ_{II} 下的机密性. 敌手如果不能够通过多项式时间的计算以不可忽略优势 $Adv^{c}(\lambda)$ 赢得参考文献[4]定义的攻击方式 λ_{II} 下机密性相关游戏，则称方案具有适应性选择密文攻击下的机密性，且该攻击类型下的攻击敌手标记为 $\lambda_{\text{II-1}}$.

定义 5. 攻击方式 λ_{I} 下的不可伪造性. 在多项

式时间内，敌手如果不能够以不可忽略优势 $Adv^{u}(\lambda)$ 赢得参考文献[4]定义的攻击方式 λ_{I} 下不可伪造性相关游戏，则称方案具有适应性选择消息攻击下的不可伪造性，且该攻击类型下的攻击敌手标记为 $\lambda_{\mathrm{I}\text{-}2}$.

定义 6. 攻击方式 λ_{II} 下的不可伪造性. 在多项式时间内，敌手如果不能够以不可忽略优势 $Adv^{u}(\lambda)$ 赢得参考文献[4]定义的攻击方式 λ_{II} 下不可伪造性相关游戏，则称方案具有适应性选择消息攻击下的不可伪造性，且该攻击类型下的敌手标记为 $\lambda_{\mathrm{II}\text{-}2}$.

4　方案步骤

4.1　签密模型

（1）注册（Setup）

密钥生成中心 KGC 根据安全参数 λ 生成大素数 p 和循环群 G_p，以及在循环群 G_p 上的椭圆曲线 $E: y^2 = x^3 + ax + b$，其中 $4a^3 + 27b^2 \neq 0$. 选取椭圆曲线 E 上的一个生成元 G，其对应的阶 q 为大素数.

系统所用的哈希函数定义如下：

$$H_1: \{0,1\}^{L_l} \times G_p \times G_p \to Z_q^*,$$
$$H_2: G_p \to \{0,1\}^{L_m + |Z_q^*|},$$
$$H_3: \{0,1\}^{L_l} \times \{0,1\}^{L_m} \times G_p \times G_p \to Z_q^*,$$
$$H_4: \{0,1\}^{L_l} \times G_p \times \{0,1\}^{L_m} \to Z_q^*,$$

其中 L_l 为用户身份 ID 的长度，L_m 为明文长度，$|Z_q^*|$ 为 Z_q^* 中数据的长度.

定义操作符如下：$\oplus$ 为异或操作，主要用于消息内容的加密和解密. $\|$ 为连接操作，主要用于在消息后附加参数信息.

KGC 随机选择系统主密钥 $s \in Z_q^*$，并得到系统公钥 $P_{pub} = sG$. KGC 公开参数信息 $Params_{pub} = \langle p, G_p, E, G, q, H_1, H_2, H_3, H_4, P_{pub}, \oplus, \| \rangle$，而保留系统主密钥信息 s.

（2）用户部分密钥生成（GenPartialKey）

用户 i 对应的 ID 信息标记为 ID_i，在生成用户 i 对应的密钥信息时，一部分由自己生成，另外一部分需要依赖 KGC 生成. 用户 i 随机选择秘密值 $x_i \in Z_p^*$，并得到 $X_i = x_i G$，然后将 ID_i 和 X_i 发送至 KGC.

KGC 收到上述信息后，为用户 i 生成另一部分密钥信息，其执行步骤如下. 首先 KGC 选择随机数 $r_i \in Z_q^*$ 作为秘密值，并计算得到 $Y_i = r_i G$ 以及 $y_i = r_i + sH_1(ID_i, X_i, Y_i)$. 然后 KGC 将 Y_i 公开，并将 y_i 通过可信秘密信道发送至用户 i.

（3）用户私钥生成（GenPrivateKey）

用户 i 收到 Y_i 和 y_i 后，需要校验信息的正确性，其校验公式为 $y_i G = Y_i + H_1(ID_i, X_i, Y_i) P_{pub}$. 如果校验失败，则意味着 KGC 发送过来的消息有误，生成用户私钥失败，用户 i 重新请求 KGC 生成部分密钥信息. 如果校验成立，用户 i 即可合成自己的私钥 $SK_i = (x_i, y_i)$.

（4）用户公钥生成（GenPublicKey）

对于用户 i，其身份信息为 ID_i，生成的私钥对为 $SK_i = (x_i, y_i)$，其对应的公钥对为 $PK_i = (X_i, Y_i)$.

为了方便论述，设定用户 a 需要共享数据给用户 b，其对应的私钥对和公钥对分别为

$$ID_a : \langle SK_a = (x_a, y_a), PK_a = (X_a, Y_a) \rangle,$$
$$ID_b : \langle SK_b = (x_b, y_b), PK_b = (X_b, Y_b) \rangle.$$

（5）签密（SignCrypt）

用户 a 发送消息 m 给用户 b，此时用户 a 需要基于用户 b 的公钥对消息 m 进行签密，用户 b 收到签密消息后，再利用自己的私钥对密文进行解密和校验.

用户 a 首先选择一个随机数 $\alpha \in Z_q^*$，并计算得到 $R = \alpha G$. 得到上述信息后，用户 a 执行如下计算得到 V 和 U：

$$h_1^b = H_1(ID_b, X_b, Y_b),$$
$$d = H_3(ID_a, m, X_a, R),$$
$$f = H_3(ID_a, m, Y_a, R),$$
$$V = \alpha\left(X_b + Y_b + h_1^b P_{pub}\right),$$
$$U = d(x_a + y_a) + \alpha f.$$

对于消息 m，用户 a 生成签密密文 $C = (m \| U) \oplus H_2(V)$，然后计算得到：

$$h = H_4(ID_a, R, C),$$
$$S = \frac{\alpha}{x_a + y_a + h}.$$

用户 a 将密文 $\Re = (S, C, h)$ 发送至用户 b.

（6）解密（UnSignCrypt）

用户 b 收到密文 $\Re = (S, C, h)$ 后，其解密流程为，

$$h_1^a = H_1(ID_a, X_a, Y_a),$$
$$R' = S\left(X_a + Y_a + h_1^a P_{pub} + hG\right),$$
$$V' = (x_b + y_b) R',$$
$$m \| U = C \oplus H_2(V').$$

用户 b 得到 $m \| U$ 后，即可从中解析出明文信息 m 以及用于校验内容完整性的辅助参数 U.

（7）签名校验（VerifySign）

用户 b 解密得到明文信息 m 后，需要对其进行校验，如果校验成功，则意味消息内容完整. 首先计算：

$$f = H_3\left(ID_a, m, Y_a, R'\right),$$
$$d' = H_3\left(ID_a, m, X_a, R'\right),$$

然后对消息内容进行校验，校验等式为 $UG = d'(X_a + Y_a + h_1^a P_{pub}) + fR'$.

4.2 基于区块链技术实现时效管理的签密模型

本文提出的基于区块链实现的时效管理方案，采用 EOS 进行时效管理，这是由于 EOS 中实现了一种称为“delayed transaction”的机制，能够使交易延迟指定时间后才能执行，中间无需人工干预，实现更新过程的可信性.

KGC 负责在 EOS 中发布智能合约Ω, 该智能合约Ω维护一个用于数据持久化的多索引表，用于存储用户 ID_i 和用户公钥等信息（标记为Θ）. 一个用户对应一个Ω，以实现对用户公钥等信息的精确控制. 该多索引表除了支持一个主索引外，还可支持多达 16 个二级索引，此外在智能合约中定义相关动作，用于实现对该索引表的增删改查功能. 需要说明的是，由于发布智能合约的动作比较耗时，在执行初始化步骤时，如果 KGC 知道参与签密过程用户的数量 M，则 KGC 预先部署 M 份智能合约Ω到区块链中，形成一个智能合约池. 后续当用户参与签密过程时，KGC 从这个智能合约池中取出一个没有使用的智能合约分配给该用户，这样可以缩减用户等待时间.

初始化阶段结束后，KGC 生成用户 ID_i 的Θ信息，再将Θ存储到智能合约的多索引表中，并设定延迟时间为 T_{exp}，即计时 T_{exp} 后，区块链自动执行删除Θ的操作. 删除Θ后，代表该用户公钥失效，在本方案中，$\Theta = \left\langle ID_i, PK_i = \left(X_i, Y_i\right), T_{exp} \right\rangle$. 需要说明的是，上述增删改功能并不会真正修改区块链中区块数据，底层区块链仍保留了历史信息. 用户 ID_i 对应的智能合约Ω以及存储的Θ信息是公开的，其他用户能够访问上述信息.

在本方案中，延迟时间等于公钥的有效期，当设定时间到来时，智能合约无需人工干预自动执行删除公钥的操作，使公钥失效. 在执行签密流程和解签密流程时，当前用户查询其他用户的智能合约Ω和Θ，如果能够查到指定用户的Θ信息，则代表其对应的公钥信息有效. 如果查询不到，说明指定用户的公钥信息已经失效，退出执行流程，智能合约伪码如过程 1 所示.

过程 1. 添加、更新和删除公钥的智能合约伪码

```
CLASS [[eosio::contract("oper_pk")]] oper_pk: public eosio::contract
{
STRUCT USER{NAME Di, STRING Xi, STRING Yi, UNSIGNED_INT Texp};
TYPEDEF eosio::multi_index <USER> DEF_ USER;
DEF_USER user;
//The function on_upsert can add or update data on the
//blockchain
on_upsert(NAME Di, STRING Xi, STRING Yi, UNSIGNED_INT Texp){
ASSERT check_authority (Di);
ITERATOR iterator1 = user.find(Di);
// add new element to blockchain
IF(iterator1== user.end()){
user. Emplace([&]( auto& row ){
    row.Di = Di; row.Xi = Xi;
    row.Yi = Yi; row.Texp = Texp;});
}
//update the specified user's information
ELSE {
user. modify ([&]( auto& row ){
    row.Di = Di; row.Xi = Xi;
    row.Yi = Yi; row.Texp = Texp;});
}

//define a transaction which will execute the erase
//operation after Texp seconds .
EOSIO::TRANSACTION t{};
t.actions.emplace_back(on_erase, std::make_tuple(user));
t.delay_sec = Texp;
t.send();
}
on_erase(NAME Di){
ASSERT check_authority (Di);
ITERATOR iterator1 = user.find(Di);
Check(iterator1!= user.end(), "Record does not exist");
user.erase(iterator1);
}
}
```

设定发送数据的用户为 a，接收数据的用户为

b，在本文提出的算法模型中，当执行签密操作时，发送者 a 需要访问两次区块链，分别校验用户 a 和用户 b 对应的公钥有效性. 当执行解签密操作时，接收者 b 只需要访问一次区块链，校验用户 b 的公钥有效性即可，不需要校验发送者公钥的有效性. 之所以在执行解签密操作时不校验发送者 a 的公钥有效性，是因为在实际场景中，发送者 a 将签密后的消息发送成功后，由于网络延迟等原因，可能会导致发送者 a 的公钥信息失效后接收者 b 才接收到密文信息，此时公钥未失效的接收者 b 仍然能够成功执行解签密操作. 需要说明的是，在执行签密或解签密流程时，用户可以自定义是否需要同时校验发送者和接收者公钥信息的有效性，以满足不同业务需求，算法流程如图 3 所示.

当用户 ID_i 的Θ信息被删除后，即公钥信息失效后，用户 ID_i 对应的智能合约Ω仍保留在区块链上，KGC 可以回收智能合约Ω，并将其重新放入到智能合约池中. 当需要重新激活用户 ID_i 的公钥信息时，KGC 可以重新在Ω中添加用户 ID_i 对应的Θ信息，使公钥重新生效，实现用户 ID_i 的重新加入功能，进而实现对用户的灵活控制.

在本方案中，公钥信息存储在区块链中，避免了公钥信息被恶意篡改的问题. 由于用户公钥信息的时效性由区块链进行维护，该维护过程由区块链自动执行，不占用物联网设备的计算资源. 此外，公钥更新过程无需人工参与，基于区块链技术，保证时效管理过程的可信性.

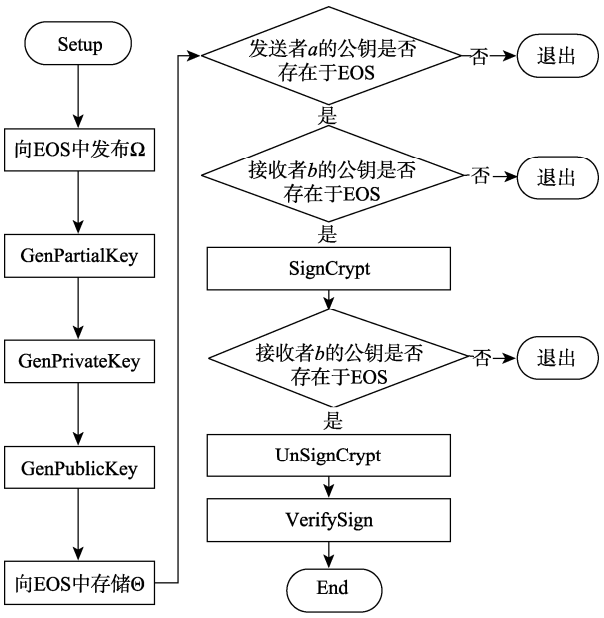

图 3　基于区块链实现时效管理的签密算法流程图

5　正确性分析

定理 1.　用户校验 KGC 为其生成的部分密钥信息是否成立的公式 $y_iG = Y_i + H_1(ID_i, X_i, Y_i)P_{pub}$ 成立.

证明.

由于 $y_i = r_i + sH_1(ID_i, X_i, Y_i)$，故可得到 $y_iG = r_iG + sH_1(ID_i, X_i, Y_i)G$.

由于 $Y_i = r_iG$，且 $P_{pub} = sG$，可得到：

$$\begin{aligned} y_iG &= r_iG + H_1(ID_i, X_i, Y_i)sG \\ &= Y_i + H_1(ID_i, X_i, Y_i)P_{pub}. \end{aligned}$$

故等式 $y_iG = Y_i + H_1(ID_i, X_i, Y_i)P_{pub}$ 成立.

定理 2.　密文接收用户 ID_b 可以从密文信息 C 中恢复明文信息 m.

证明.

密文发送者 ID_a 首先对明文信息 m 进行签密，公式为 $C = (m\|U) \oplus H_2(V)$，其中 $V = \alpha\left(X_b + Y_b + h_1^b P_{pub}\right)$.

可以得到

$$\begin{aligned} &R' = S\left(X_a + Y_a + h_1^a P_{pub} + hG\right), \\ &h = H_4(ID_a, R, C), \\ &S = \frac{\alpha}{x_a + y_a + h}, \\ &R = \alpha G. \end{aligned}$$

因此可以得到

$$\begin{aligned} R' &= \left(\frac{\alpha}{x_a + y_a + h}\right)\left(x_aG + r_aG + h_1^a sG + hG\right) \\ &= \left(\frac{\alpha}{x_a + y_a + h}\right)\left(x_a + r_a + h_1^a s + h\right)G \\ &= \left(\frac{\alpha}{x_a + y_a + h}\right)\left(x_a + y_a + h\right)G \\ &= \alpha G \\ &= R. \end{aligned}$$

由于 $V' = (x_b + y_b)R'$，故可知

$$\begin{aligned} V' &= (x_b + y_b)R \\ &= (x_b + y_b)\alpha G \\ &= \alpha(x_bG + y_bG) \\ &= \alpha(X_b + y_bG). \end{aligned}$$

由于

$$\begin{aligned} &y_i = r_i + sH_1(ID_i, X_i, Y_i), \\ &h_1^b = H_1(ID_b, X_b, Y_b), \\ &V = \alpha\left(X_b + Y_b + h_1^b P_{pub}\right), \end{aligned}$$

故可知

$$\begin{aligned}V' &= \alpha\left(X_b + y_b G\right)\\ &= \alpha\left(X_b + r_b G + H_1\left(ID_b, X_b, Y_b\right) sG\right)\\ &= V.\end{aligned}$$

而 $m\|U = C \oplus H_2(V) = C \oplus H_2(V')$，故密文接收用户 ID_b 可以获得明文消息 m，原式得证.

定理 3. 密文接收用户 ID_b 对解密出的明文信息 m 进行校验，其校验公式 $UG = d'(X_a + Y_a + h_1^a P_{pub}) + fR'$ 成立.

证明.

由于 $U = d\left(x_a + y_a\right) + \alpha f$，故可得到 $UG = d\left(x_a G + y_a G\right) + f\alpha G$.

因为 $y_a = r_a + sH_1\left(ID_a, X_a, Y_a\right)$，故得到 $UG = d\left(x_a G + r_a G + H_1\left(ID_a, X_a, Y_a\right) sG\right) + f\alpha G$.

由于 $R = \alpha G = R'$，$X_a = x_a G$，$Y_a = r_a G$，$P_{pub} = sG$，$h_1^a = H_1\left(ID_a, X_a, Y_a\right)$，故可知

$$\begin{aligned}UG &= d\left(x_a G + r_a G + H_1\left(ID_a, X_a, Y_a\right) sG\right) + f\alpha G\\ &= d\left(X_a + Y_a + h_1^a P_{pub}\right) + fR'.\end{aligned}$$

由于 $d' = H_3\left(ID_a, m, X_a, R\right) = d$，故可知 $UG = d'\left(X_a + Y_a + h_1^a P_{pub}\right) + fR'$，原式得证.

6 安全性分析

6.1 机密性

定理 4. 在攻击方式 λ_{I} 下本方案能够保证系统的机密性，即敌手 $\lambda_{\mathrm{I\text{-}1}}$ 不能够通过多项式时间的计算以不可忽略优势 $Adv^c(\lambda)$ 赢得本次游戏.

证明. 详细证明见附录.

定理 5. 攻击方式 λ_{II} 下，本方案能够保证系统的机密性. 即在随机预言机模型下，敌手 $\lambda_{\mathrm{II\text{-}1}}$ 最终以不可忽略优势 ε 在多项式时间内赢得相关游戏，如果敌手最多进行 ς 次签密查询和 ξ 次私钥生成查询，则系统能以不可忽略优势 $Adv^c(\lambda) \geqslant \left(1 - \frac{\xi}{2^{\lambda}}\right)^2 \frac{\varepsilon}{e(\varsigma + 1)}$ 在多项式时间内解决 CDH 问题.

证明. 详细证明见附录.

6.2 不可伪造性

定理 6. 攻击方式 λ_{I} 下本方案能够保证系统的不可伪造性，即敌手 $\lambda_{\mathrm{I\text{-}2}}$ 不能够通过多项式时间的计算以不可忽略优势 $Adv^u(\lambda)$ 赢得本次游戏.

证明. 详细证明见附录.

定理 7. 攻击方式 λ_{II} 下本方案能够保证系统的不可伪造性. 敌手 $\lambda_{\mathrm{II\text{-}2}}$ 不能够通过多项式时间的计算以不可忽略优势 $Adv^u(\lambda)$ 赢得本次游戏. 即在随机预言机模型下，敌手 $\lambda_{\mathrm{II\text{-}2}}$ 最终以不可忽略优势 ε 在多项式时间内赢得相关游戏，如果敌手最多进行 ς 次签密查询和 ξ 次私钥生成查询，则系统能以不可忽略优势 $Adv^u(\lambda) \geqslant \left(1 - \frac{\xi}{2^{\lambda}}\right) \frac{\varepsilon}{e(\varsigma + 1)}$ 在多项式时间内解决 DL 问题.

证明. 详细证明见附录.

6.3 密钥托管

在本文中，KGC 负责用户部分密钥生成操作，用户将 $\langle ID_i, X_i \rangle$ 发送至 KGC，KGC 基于用户的身份信息 ID_i 和公开参数 X_i 为用户生成部分密钥信息 y_i 和其对应的部分公钥信息 Y_i. 若 KGC 想获取用户的私钥信息 $SK_i = (x_i, y_i)$，则只能从 X_i 中获取 x_i 信息，而求解该问题的难度等价于求解 DL 问题，即 KGC 并不能从自己掌握的信息中推断出用户的私钥信息，故本方案能够实现密钥安全托管.

6.4 公开验证性

当签密发送者和签密合成者之间对密文 $\Re = (S, C, h)$ 存在异议时，只需要将签密发送者的身份信息 ID_a、公钥信息 $PK_a = (X_a, Y_a)$、$\mathrm{M} = (x_b + y_b)$ 等其它公开信息发送至校验方，校验方计算得到 $R' = S\left(X_a + Y_a + h_1^a P_{pub} + hG\right)$，$V' = \mathrm{M}R'$，进而通过 $m'\|U = C \oplus H_2(V')$ 解密出明文信息 m'. 利用公开信息分别计算出 $h_1^a = H_1\left(ID_a, X_a, Y_a\right)$，$f = H_3(ID_a, m, Y_a, R')$，$d' = H_3\left(ID_a, m, X_a, R'\right)$，然后校验 $UG = d'(X_a + Y_a + h_1^a P_{pub}) + fR'$. 从上述过程可见，校验过程不需要使用私密信息，因此该方法具有公开验证性.

6.5 不可否认性

由定理 3 和定理 4 可知，本文方法对于攻击方式 λ_{I} 和攻击方式 λ_{II} 具有不可伪造性，也就意味第三方无法伪造密文信息，签密合成者无法对自己生成的密文进行否认，而且由于本方法具有公开验证性，任意第三方均可以对密文信息进行校验，因此该方法具有不可否认性.

6.6 匿名性

在实际应用场景中，第三方可以捕获密文信息，将密文信息与用户身份信息关联起来，并执行基于身份的流量分析. 在本文方法中，签名发送者在进行签密时，首先会选择一个随机数 $\alpha \in Z_q^*$，并计算得到 $R = \alpha G$，然后基于 R 计算签密信息，即 $h = H_4\left(ID_a, R, C\right)$，$S = \alpha / \left(x_a + y_a + h\right)$，$V = \alpha(X_b + Y_b +$

$h_1^b P_{pub}$)，得到密文 $C=(m\|U)\oplus H_2(V)$. 由于每一次签密过程的 α 均不相同，导致 $\Re=(S,C,h)$ 中每个参数均不相同，而且发送的 $\Re$ 并不含有用户身份信息，而且无法从中计算得到用户身份信息，因此本方案具有匿名性.

7　性能分析

本部分将从安全属性和计算复杂度两个方面对签密算法进行考察，并给出密文长度的对比结果.

7.1　安全属性分析

在对密文长度进行对比时，定义了如下符号：其中 $|m|$ 代表了明文长度，$|G_p|$ 代表了群 G_p 上相关元素的长度，$|Z_q^*|$ 代表了 Z_q^* 上相应元素的长度，$|ID|$ 代表用户 ID 的长度.

本文主要选择了基于离散对数签密算法[8-9, 29]、基于双线性映射签密算法[30]为对比算法，主要对比其密文长度和安全属性. 基于双线性映射的方案其计算复杂度较高，如果待签密信息的数据量较大，将会影响算法的执行效率[31].

当前相关研究人员提出了一些抗量子攻击的签密方案，本文选择了两种抗量子攻击的签密方法作为对比算法，分别是基于编码密码体制的签密算法和基于多变量密码体制的签密算法，主要侧重对比其安全属性. 由于抗量子攻击的签密算法实现原理与本文算法差异大，密文长度依赖项不同，无法对其进行统一对比，故本文主要侧重对比其安全属性，不再考察其密文长度. 在安全属性方面，与现有方案的对比结果如表 1 所示.

表 1　安全属性对比结果

签密方案	实现方式	密文长度	不可伪造性	机密性	公开验证性	不可否认性	匿名性	密钥安全托管
文献[11]	编码密码体制	×	Yes	Yes	No	Yes	Yes	No
文献[18]	多变量密码体制	×	Yes	Yes	No	Yes	Yes	No
文献[8]	离散对数	$\|m\|+3\|Z_q^*\|$	Yes	Yes	Yes	Yes	Yes	Yes
文献[29]	离散对数	$\|m\|+2\|Z_q^*\|$	Yes	Yes	No	Yes	Yes	Yes
文献[9]	离散对数	$\|m\|+2\|Z_q^*\|$	No	No	No	No	Yes	Yes
文献[30]	双线性对	$\|m\|+3\|G_p\|+\|Z_q^*\|$	Yes	Yes	Yes	Yes	Yes	Yes
本文	椭圆曲线	$\|m\|+3\|Z_q^*\|$	Yes	Yes	Yes	Yes	Yes	Yes

抗量子攻击的签密算法，例如文献[11]和文献[18]，均不能实现公开验证性和密钥安全托管. 虽然上表中基于离散对数的实现方案密文长度普遍较短，但是在达到同等级别的安全性要求下，其密钥长度较长，存储密钥占用的存储空间较多. 此外，虽然文献[8]的安全属性和密文长度与本文相似，但是在执行签密和解签密流程时，其计算复杂度要高于本文算法. 文献[30]基于双线性对进行实现，具有较高的安全强度，然而该算法的密文长度比本文要长.

7.2　计算复杂度分析

本文提出的签密算法，基于椭圆曲线进行实现，与基于离散对数和基于大数分解的签密方案相比，能够以较小的密钥长度实现同等强度的安全保护. 为了便于对比计算复杂度，表 2 定义了相关符号，根据文献[32]，本文给出了常见运算符换算成模乘运算符的比率，以便于算法对比. 此外需要说明的是，由于模加法、模减法以及椭圆曲线加法计算开销较低，这里不对其进行考察，对比结果如表 3 所示.

表 2　签密方案符号表示

符号	定义	描述	符号	定义	描述
T_m	模乘运算	基准运算	T_{pm}	椭圆曲线点乘运算	$T_{pm}\approx 29T_m$
T_e	模幂运算	$T_e\approx 240T_m$	T_h	哈希运算	×
T_b	双线性对运算	$T_b\approx 87T_m$			

表 3 中选择的算法均是无证书可公开验证的签密算法，其中文献[8-9, 29]是基于离散对数进行实现，而文献[30]是基于双线性对进行实现. 基于椭圆曲线的实现方案，在达到同样强度的安全等级要求时，所需的密钥长度一般较小，相比较采用双线性对的操作，其执行效率较高. 从上述对比结果可知，本文算法均比其它三种算法[8-9, 29]计算复杂度低. 文献[30]基于离散对数实现签密步骤，在解签密时，需要依赖双线性对进行实现，而且需要执行一次求逆运算，因此计算复杂度比本文要高.

表 3 计算复杂度对比

方案名称	签密		解签密	
	计算复杂度	统一化	计算复杂度	统一化
本文算法	$3T_{pm}+4T_h+3T_m$	$\approx 90T_m+4T_h$	$6T_{pm}+3T_h$	$\approx 174T_m+3T_h$
文献[8]	$3T_e+4T_h+5T_m$	$\approx 725T_m+4T_h$	$6T_e+4T_h+4T_m$	$\approx 1444T_m+4T_h$
文献[29]	$3T_m+4T_e+4T_h$	$\approx 963T_m+4T_h$	$5T_e+5T_h+5T_m$	$\approx 1205T_m+5T_h$
文献[9]	$2T_h+3T_e+3T_m$	$\approx 723T_m+2T_h$	$2T_h+3T_m+4T_e$	$\approx 963T_m+2T_h$
文献[30]	$4T_e+T_h+4T_m$	$\approx 964T_m+T_h$	$2T_e+T_h+3T_m+2T_b$	$\approx 657T_m+T_h$

8 仿真实验

8.1 签密方案性能分析

本文椭圆曲线参数设置参考了 Secp256k1，即比特币采用的参数[33]. Secp256k1 是经优化的基于有限域 $\mathbb{F}_p$ 的椭圆曲线，与其它曲线相比，其性能能够提升 30%左右. 由于其密钥长度较短，占用较小的存储和带宽资源，能够有效适配到区块链应用场景中. 对于椭圆曲线 $E(\mathbb{F}_p)$: $y^2 \equiv x^3+ax+b(\mathrm{mod}\ p)$，$a,b\in\mathbb{F}_p$，生成元为 G，其对应的阶为 n，参数设置如表 4 所示，实验环境配置参数如表 5 所示.

设定两个用户参与数据签密过程，分别为用户 a 和用户 b，首先生成用户 a 和 b 的私钥信息和公钥信息，用户 a 对数据 m 进行签密，并将密文发送给用户 b 进行解签密，校验成功后，显示发送成功的消息.

表 4 椭圆曲线参数设置

$p=0x$FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFEFFFFFC2F

$a=0x0$

$b=0x7$

$G=\begin{pmatrix}0x79BE667EF9DCBBAC55A06295CE870B07029BFCDB2DCE28D959F2815B16F81798,\\ 0x483ADA7726A3C4655DA4FBFC0E1108A8FD17B448A68554199C47D08FFB10D4B8\end{pmatrix}$

$n=0x$FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFEBAAEDCE6AF48A03BBFD25E8CD0364141

表 5 执行环境配置

硬件环境:	
CPU:	Intel（R） Core（TM） i5-7200U 2.40GHz
RAM:	8.00GB
磁盘:	SATA 1TB
软件环境:	
Windows:	Windows 7 64
MyEclipse:	2015 CI
JAVA:	1.7
密码库：	
JPBC	2.0.0

本部分主要从初始化、密钥生成、签密和解签密四个部分考察签密方案的性能，其中初始化部分即本文方案第一步的注册过程，密钥生成部分包括了用户部分密钥生成、用户私钥生成以及用户公钥生成三个步骤，签密部分即本方案中的签密步骤，解签密部分包括了解密和签名校验两个步骤. 为了验证方案的执行效率，设定 200 个用户参与签密，其中 100 个用户作为发送者，其他 100 个用户作为接收者，并统计执行上述 4 个步骤的执行时间平均值. 本文选择了 Karati[30]和 Wang[3]的方案作为对比算法，图 4 给出了发送 20Byte 数据时，签密算法四个步骤执行时间，图 5 和图 6 给出了签密步骤和解签密步骤执行时间与发送数据量之间的关系.

由图 5 和图 6 可知，本文算法解签密步骤执行时间比签密步骤要长，这是因为解签密包括了解密和校验两个过程，从图 4 可知解签密的时间大概是签密时间的 3.8 倍. 表 3 中新方案中的解签密步骤的计算复杂度大概为 $174T_m+3T_h$，而签密为 $90T_m+4T_h$，

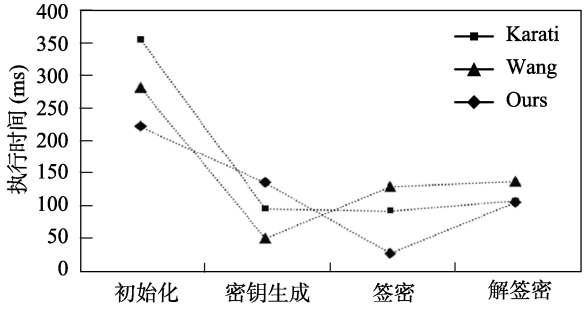

图 4 签密算法各阶段执行时间

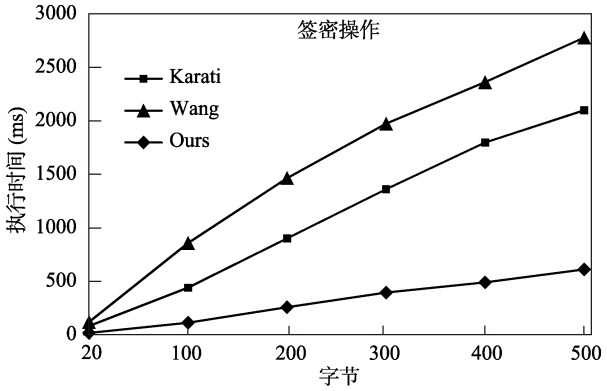

图 5　签密步骤执行时间

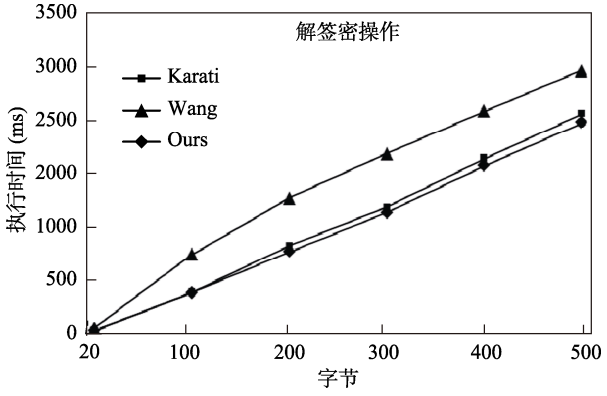

图 6　解签密步骤执行时间

假设 T_h 与 T_m 执行时间相近，解签密操作大概是签密操作执行时间的 2 倍左右，这与仿真实验结果接近.

从图 5 和图 6 可知，在签密和解签密阶段，本文算法整体上要优于对比算法. 在密钥生成阶段，Wang 算法优于本文算法，这是由于 Wang 的算法基于离散对数生成密钥信息，执行效率较高，但是在解签密时，需要依赖双线性对实现数据校验，要比本文算法执行效率低.

从图 4 可知，本文算法执行解签密步骤时所耗时间要比签密步骤长，当对解签密步骤进行优化时，有两种可行的优化方向. 第一个优化方向是当用户 a 向用户 b 发送多条数据时，对于 $h_1^a = H_1(ID_a, X_a, Y_a)$，只需要执行一次解签密步骤即可. 第二是在发送多条数据时，对于 $R' = S\left(X_a + Y_a + h_1^a P_{pub} + hG\right)$，只需要计算一次 $X_a + Y_a + h_1^a P_{pub}$ 即可.

8.2　基于区块链实现时效管理的签密算法性能分析

将本文签密方案适配到区块链应用场景中时，执行初始化步骤时 KGC 需要将智能合约Ω发布到区块链中. 在生成用户公钥信息后，KGC 需要将 $\Theta = \left\langle ID_i, PK_i = (X_i, Y_i), T_{exp} \right\rangle$ 存储到区块链中. 在执行签密流程时，用户需要访问区块链查询发送者和接收者的公钥信息. 在执行解签密流程时，用户需要访问区块链查询接收者的公钥信息，本部分将对引入区块链后导致的性能损失进行评估.

区块链采用 EOS，相比较其它区块链项目，EOS 能够支持数百万级别的用户，而且具有强大的并行执行能力[34]. 基于 EOS 搭建区块链仿真环境，配置参数如表 6 所示. 设定只有两个用户参与签密过程，一个用户为发送者，另外一个为接收者，KGC 发布两份智能合约，用于存储用户的公钥等信息，执行时间结果如图 7 所示. 设定原始签密算法执行某一步骤时的平均执行时间为 T，基于区块链实现时效管理的签密算法执行同一步骤时的平均执行时间为 $\overline{t}$，则在执行这一步骤时候的性能损失百分比为 $|\overline{t}\text{-T}|/\text{T}\times 100\%$，实验结果如图 8 所示.

表 6　区块链环境配置

硬件环境：	
阿里云服务器 ECS:	3
CPU:	Intel Xeon E5-2682 v4
RAM:	2.00 GB
带宽:	1.00 MB
SATA 大小:	40GB
软件环境：	
Ubuntu:	16.04.6
Eosio:	1.7.0
Eosio.cdt:	1.6.1
Eosio.contracts:	1.5.2
JAVA:	1.7
JPBC	2.0.0

从图 7 和图 8 可知，初始化步骤相比较其它 3 个步骤，耗时较多，这是由于在执行初始化步骤时，预先发布了两份智能合约到 EOS 中，后续用户执行密钥生成操作后，只需要将公钥等信息存储到区块链上即可，不需要执行耗时较多的智能合约发布过程. 从实验结果可知，发布智能合约的操作，大概耗时 150 ms，整个初始化过程占用了不到 400 ms 的时间. 对于在日常业务场景中执行次数较多的签密和解签密操作，签密操作耗时约为 30 ms，解签密操作耗时约为 100 ms，因此能够适配到大部分的物联网场景中.

从图 8 可知，签密过程性能损失为 7%，而解签密步骤性能损失不到 1%，这是由于签密过程需要访问两次区块链来获取公钥信息，而解签密只需要访问一次. 由此可见，引入区块链后，对签密步骤影响较大. 对于初始化过程，由于需要部署智能合约，会导致性能损失 70%左右，但是由于只需要执行一次初始化步骤，在系统整个生命周期内，初始化步

骤对系统整体性能影响较低.

用户公钥信息存储在区块链中，基于区块链不可篡改特性保证了公钥信息的安全性. 此外由于用户公钥信息的时效性由区块链进行维护，不占用物联网计算资源，因此本文方案能够有效适配到物联网应用场景中.

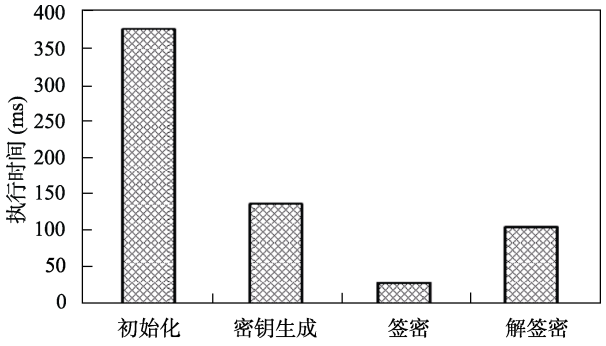

图 7 基于区块链实现时效管理的签密算法执行时间

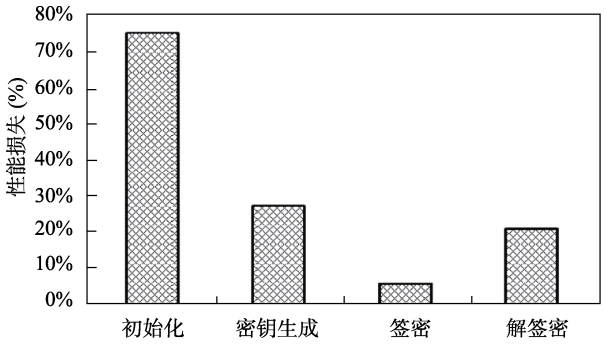

图 8 基于区块链实现时效管理的签密算法性能损失

9 结 论

签密算法能够同时实现消息机密性和完整性，本文基于椭圆曲线提出一种无证书签密方案，新方案具有公开验证性、机密性和不可伪造性等安全属性. 提出一种基于区块链的时效管理方案，由智能合约按照时间戳更新公钥有效性，基于区块链的特性保证了公钥的不可篡改性以及更新过程的可信性. 新方案具有较短的密钥长度，且具有较高的运行效率，能够有效适配到物联网场景中.

致谢 感谢评审专家及编辑老师对稿件进行的细致审阅，感谢郑州师范学院付俊俊和周春天对论文格式提出的修改意见!

参 考 文 献

[1] Zheng YL. Digital signcryption or how to achieve cost (signature & encryption)≪ cost (signature)+ cost (encryption)//Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology. Santa Barbara, USA, 1997: 165-179

[2] Li MR. A blockchain based security incentive mechanism in crowdsensing applications[M.S. Thesis]. North China University of Technology, Beijing, 2019 (in Chinese)
(李梦茹. 群智感知中基于区块链的安全激励机制研究[硕士学位论文].北方工业大学, 北京, 2019)

[3] Wang JZ, Li MR, He YH, et al. A blockchain based privacy-preserving incentive mechanism in crowdsensing applications. IEEE Access, 2018, 6(1): 17545-17556.

[4] Sattam SA-R, Kenneth GP. Certificateless public key cryptography//Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security. Taipei, China, 2003:452-473

[5] Barbosa M, Farshim P. Certificateless signcryption//Proceedings of the 2008 ACM Symposium on Information, Computer and Communications Security. Tokyo, Japan, 2008:369-372

[6] Li FG, Masaaki S, Tsuyoshi T. Certificateless hybrid signcryption//Proceedings of the 5th International Conference on Information Security Practice and Experience. Tokyo, Japan, 2009:112-123

[7] Wu CH, Chen ZX. A new efficient certificateless signcryption scheme//Proceedings of the 2008 International Symposium on Information Science and Engineering. Shanghai, China, 2008: 661-664

[8] Zhou YW, Yang B, Zhang WZ. Security analysis and improvement of certificateless signcryption scheme without bilinear pairing. Chinese Journal of Computers, 2016, 39(6): 1257-1266(in Chinese)
(周彦伟, 杨波, 张文政. 不使用双线性映射的无证书签密方案的安全性分析及改进. 计算机学报, 2016, 39(6): 1257-1266)

[9] Zhu H, Li H, Wang YM. Certificateless signcryption scheme without pairing. Journal of Computer Research and Development, 2010, 47(9): 1587-1594(in Chinese)
(朱辉, 李晖, 王育民. 不使用双线性对的无证书签密方案. 计算机研究与发展, 2010, 47(9): 1587-1594.)

[10] Li JM, Yu HF, Xie Y. EIGamal broadcasting multi-signcryption protocol with UC security. Journal of Computer Research and Development, 2019, 56(5): 1101-1111(in Chinese)
(李建民, 俞惠芳, 谢永. 通用可复合的 ElGamal 型广播多重签密协议. 计算机研究与发展, 2019, 56(5): 1101-1111)

[11] Wang Z, Han YL. Anti-quantum signcryption scheme based on Niederreiter cryptosystem. Computer Engineering, 2020, 46(5): 193-199 (in Chinese)
(王众, 韩益亮. 基于 Niederreiter 密码体制的抗量子签密方案. 计算机工程, 2020, 46(5): 193-199)

[12] Li ZH, Yang YT, Li ZC. New public key cryptography based on QC-MDPC code. Application Research of Computers, 2015, 32(3): 881-884(in Chinese)
(李泽慧, 杨亚涛, 李子臣. 基于 QC-MDPC 码的公钥密码方案设计. 计算机应用研究, 2015, 32(3): 881-884.)

[13] Cao D, Zhao SM, Song YL. Quantum McEliece public key cryptosystem based on quantum QC-LDPC codes. Journal of Nanjing University of Posts and Telecommunications(Natural Science), 2011, 31(2): 64-68(in Chinese)
(曹东, 赵生妹, 宋耀良. 一种基于量子准循环 LDPC 码的 McEliece 公钥密码算法. 南京邮电大学学报(自然科学版),

2011, 31(2): 64-68)
[14] Wang YL. Research on McEliece public-key cryptosystem based on QC-LDPC code[M.S. Thesis]. Xidian University, Xian, 2013(in Chinese)
(王延丽. 基于 QC-LDPC 码的 McEliece 公钥密码体制研究[硕士学位论文]. 西安电子科技大学, 西安, 2013)
[15] Lu XH. Research on signature and signcryption schemes based on Lattice problems[M.S. Thesis]. Beijing University of Posts and Telecommunications, BeiJing, 2019
(路秀华. 基于格问题的签名和签密方案研究[硕士学位论文]. 北京邮电大学, 北京, 2019)
[16] Zhao XF, Wang X. An efficient identity-based signcryption from lattice. International Journal of Security and Its Applications, 2014, 8(2): 363-369
[17] Yan JH, Wang LC, Dong MX, et al. Identity‐based signcryption from lattices. Security and Communication Networks, 2015, 8(18): 3751-3770
[18] Han YL, Lan JJ, Yang XY. A signcryption scheme based on LRPC and multivariate cryptosystem. Journal of Cryptologic Research, 2016, 3(1): 56–66(in Chinese)
(韩益亮, 蓝锦佳, 杨晓元. 基于 LRPC 码和多变量的签密方案. 密码学报, 2016, 3(1): 56-66)
[19] Tao CD, Diene A, Tang SH, et al. Simple matrix scheme for encryption//Proceedings of the International Workshop on Post-Quantum Cryptography. Limoges, France, 2013: 231-242
[20] Jaiberth P, John B, Ding JT. ZHFE, A new multivariate public key encryption scheme//Proceedings of the International Workshop on Post-quantum Cryptography. Waterloo, Canada, 2014:229-245
[21] Ding JT, Albrecht P, Wang L-c. The cubic simple matrix encryption scheme//Proceedings of the International Workshop on Post-quantum Cryptography. Waterloo, Canada, 2014: 76-87
[22] The Case Against Quantum Computing, https://spectrum.ieee.org/computing/hardware/the-case-against-quantum-computing 2018, 11, 15
[23] Wang LP, Hu MS, Jia ZJ, et al. A voting scheme in blockchain based on threshold group signature. Springer Communications in Computer and Information Science, 2018, 960(1): 184-202
[24] Qian WN, Shao QF, Zhu YC, et al. Research problems and methods in blockchain and trusted data management. Journal of Software, 2018, 29(1): 150-159(in Chinese)
(钱卫宁, 邵奇峰, 朱燕超等. 区块链与可信数据管理:问题与方法. 软件学报, 2018, 29(1): 150-159)
[25] Zhu LH, Gao F, Shen M, et al. Survey on privacy preserving techniques for blockchain technology. Journal of Computer Research and Development, 2017, 54(10): 2170-2186(in Chinese)
(祝烈煌, 高峰, 沈蒙等. 区块链隐私保护研究综述. 计算机研究与发展, 2017, 54(10): 2170-2186)
[26] Wang LP, Hu MS, Jia ZJ, et al. A signature scheme applying on blockchain voting scene based on Asmuth-Bloom algorithm//Proceedings of the IEEE 4th International Conference on Computer and Communications. Chengdu, China, 2018:2372-2378.
[27] Ma ZF, Wang XC, Jain DK, et al. A blockchain-based trusted data management scheme in edge computing. IEEE Transactions on Industrial Informatics, 2020, 16(3): 2013 - 2021
[28] Shao QF, Jin CQ, Zhang Z, et al. Blockchain: architecture and research progress. Chinese Journal of Computers, 2018, 41(5): 969-988(in Chinese)
(邵奇峰, 金澈清, 张召等. 区块链技术:架构及进展. 计算机学报, 2018, 41(5): 969-988)
[29] Zhao ZG. Security analysis and improvement of a certificateless signcryption scheme. Journal on Communications, 2015, 36(3): 129-134(in Chinese)
(赵振国. 无证书签密机制的安全性分析与改进. 通信学报, 2015, 36(3): 129-134)
[30] Arijit K, Hafizul IS, GP B, et al. Provably secure identity-based signcryption scheme for crowdsourced industrial. Internet of Things environments. IEEE Internet of Things Journal, 2018, 5(4): 2904-2914
[31] Zhou YW, Yang B, Zhang WZ. Provably secure and efficient leakage-resilient certificateless signcryption scheme without bilinear pairing. Discrete applied mathematics, 2016, 204(1): 185-202
[32] Jia SP. Research of certificateless anonymous multi-receiver signcryption based on ECC[M.S. Thesis]. Xidian University, Xian, 2018 (in Chinese)
(贾生盼. 基于椭圆曲线的无证书匿名多接收者签密研究[硕士学位论文]. 西安电子科技大学,西安,2018)
[33] SEC 2: Recommended elliptic curve domain parameters, http://www.secg.org/sec2-v2.pdf 2010,1,27
[34] Li WC, Lin QJ, Guo QK, et al. Dive into EOS: theory analysis and developmental Practices. Beijing: China Machine Press, 2019(in Chinese)
(李万才, 林琪钧, 郭琦康等. 深入理解 EOS:原理解析与开发实战. 北京: 机械工业出版社, 2019)

附 录

定理 4. 在攻击方式 λ_{I} 下本方案能够保证系统的机密性，即敌手 $\lambda_{\mathrm{I\text{-}1}}$ 不能够通过多项式时间的计算以不可忽略优势 $Adv^{c}(\lambda)$ 赢得本次游戏.

证明. 对于攻击方式 λ_{I}，公钥信息存储在区块链中，由于区块链具有不可篡改的特性，敌手 $\lambda_{\mathrm{I\text{-}1}}$ 并不能够通过篡改公钥来发起对系统的攻击，敌手 $\lambda_{\mathrm{I\text{-}1}}$ 不能够通过多项式时间的计算以不可忽略优势 $Adv^{c}(\lambda)$ 赢得本次游戏. 因此对于攻击方式 λ_{I}，系统方案具有机密性.

定理 5. 攻击方式 λ_{II} 下，本方案能够保证系统的机密性. 即在随机预言机模型下，如果敌手 $\lambda_{\mathrm{II\text{-}1}}$ 最多能够执行 ς 次签密查询操作以及 ξ 次私钥生成查询操作，就能够在多项式时间内以不可忽略优势 ε 赢得相关游戏，那么系统就能以不可忽略优势 $Adv^{c}(\lambda)\geqslant\left(1-\frac{\xi}{2^{\lambda}}\right)^{2}\frac{\varepsilon}{e(\varsigma+1)}$ 在多项式时间内解决 CDH 问题.

证明. 令算法 Γ 为 CDH 问题的解决算法，设定输入为 $\langle G,aG,bG\rangle$，其中 $a\in Z_p^*$，$b\in Z_p^*$，而且其数值未知，算法的目标就是求解 abG. 在游戏中敌手 $\lambda_{\mathrm{II\text{-}1}}$ 充当挑战

者身份，Γ 利用敌手攻击系统的过程，来求解上述 CDH 问题. 首先 Γ 通过执行 Setup 步骤来初始化系统环境，得到参数 $Params_{pub}=\langle p,G_p,E,G,q,H_1,H_2,H_3,H_4,P_{pub},\oplus,\|\rangle$，其中 $P_{pub}=aG$，并将 $Params_{pub}$ 发送至敌手 $\mathcal{A}_{\text{II-1}}$，且系统主密钥 a 不被敌手 $\mathcal{A}_{\text{II-1}}$ 所知. 维护列表 L_1、L_2、L_3、L_4、L_{sk} 和 L_{pk}，这些列表用于存储敌手 $\mathcal{A}_{\text{II-1}}$ 对预言机 H_1、H_2、H_3、H_4 以及私钥生成操作和公钥生成操作的询问结果，开始执行下述操作前，首先将上述列表初始化为空.

询问阶段：

H_2 询问：当 Γ 收到敌手 $\mathcal{A}_{\text{II-1}}$ 对 H_2 询问 $H_2(V)$ 的信息时，若存在 $\langle V,h_2\rangle\in L_2$，则返回相应的 h_2 给敌手 $\mathcal{A}_{\text{II-1}}$，否则 Γ 选取满足条件 $\langle V,h_2\rangle\notin L_2$ 的随机数 h_2，并添加 $\langle V,h_2\rangle$ 到 L_2 中.

H_3 询问：当 Γ 收到敌手 $\mathcal{A}_{\text{II-1}}$ 对 H_3 询问 $H_3(ID_i,m,X(Y),R)$ 的信息时，若存在 $\langle ID_i,m,X(Y),R,h_3\rangle\in L_3$，则返回相应的 h_3 给敌手 $\mathcal{A}_{\text{II-1}}$，否则 Γ 选取满足条件 $\langle ID_i,m,X(Y),R,h_3\rangle\notin L_3$ 的随机数 h_3，并添加 $\langle ID_i,m,X(Y),R,h_3\rangle$ 到 L_3 中.

H_4 询问：当 Γ 收到敌手 $\mathcal{A}_{\text{II-1}}$ 对 H_4 询问 $\langle ID_i,R,C\rangle$ 的信息时，若存在 $\langle ID_i,R,C,h_4\rangle\in L_4$，则返回相应的 h_4 给敌手 $\mathcal{A}_{\text{II-1}}$，否则 Γ 选取满足条件 $\langle ID_i,R,C,h_4\rangle\notin L_4$ 的随机数 h_4，并添加 $\langle ID_i,R,C,h_4\rangle$ 到 L_4 中.

公钥生成询问：当 Γ 收到敌手 $\mathcal{A}_{\text{II-1}}$ 对 ID_i 的公钥生成询问时，Γ 会查询列表 L_{pk}，如果列表 L_{pk} 存在元素 $\langle ID_i,X_i,Y_i,c_i\rangle$，其中 c_i 只可以为 0 或 1，则返回公钥信息 PK_i 给敌手 $\mathcal{A}_{\text{II-1}}$；否则 Γ 随机为 c_i 进行赋值，且设定 $\Pr[c_i=1]=\delta$，其中 $\delta=\dfrac{1}{\varsigma+1}$. 若 $c_i=0$，Γ 选取满足 $\langle ID_i,X_i,Y_i,c_i\rangle\notin L_{pk}$ 的不大于 p 随机整数 x_i,y_i,h_1，其中 $X_i=x_iG$，$Y_i=y_iG-h_1P_{pub}$. 然后将公钥信息 PK_i 发送给敌手 $\mathcal{A}_{\text{II-1}}$，并执行 $L_{pk}\leftarrow\langle ID_i,X_i,Y_i,c_i\rangle$，$L_{sk}\leftarrow\langle ID_i,x_i,y_i\rangle$，$L_1\leftarrow\langle ID_i,X_i,Y_i,h_1\rangle$. 若 $c_i=1$，令 $X_i=x_i^{know}G$，$Y_i=r_i^{know}G$，其中 $x_i^{know},r_i^{know}\in Z_p^*$，$\langle ID_i,X_i,Y_i,c_i\rangle\notin L_{pk}$. 选取满足条件 $\langle ID_i,x_i^{know},y_i\rangle\notin L_{sk}$，$\langle ID_i,X_i,Y_i,h_1\rangle\notin L_1$ 和 $Y_i=y_iG-h_1P_{pub}$ 的随机数 y_i 和 h_1，且均为 Γ 知晓. $L_{pk}\leftarrow\langle ID_i,X_i,Y_i,c_i\rangle$，$L_{sk}\leftarrow\langle ID_i,x_i,y_i\rangle$，$L_1\leftarrow\langle ID_i,X_i,Y_i,h_1\rangle$，并返回公钥信息 PK_i 给敌手 $\mathcal{A}_{\text{II-1}}$.

H_1 询问：当 Γ 收到敌手 $\mathcal{A}_{\text{II-1}}$ 询问 H_1 信息时，若存在 $\langle ID_i,X_i,Y_i,h_1\rangle\in L_1$，则返回相应的 h_1 给敌手 $\mathcal{A}_{\text{II-1}}$，否则 Γ 对 ID_i 进行公钥生成询问后，返回 L_1 中的 h_1 给敌手 $\mathcal{A}_{\text{II-1}}$.

私钥生成询问：当 Γ 收到敌手 $\mathcal{A}_{\text{II-1}}$ 对 ID_i 询问私钥信息时，若存在 $\langle ID_i,x_i,y_i\rangle\in L_{sk}$，则返回 SK_i 给敌手 $\mathcal{A}_{\text{II-1}}$，否则执行公钥生成询问，并从 L_{sk} 中查找相应的元素 $\langle ID_i,x_i,y_i\rangle$，并返回私钥信息 SK_i.

签密询问：设发送端为 ID_s，接收端为 ID_r，敌手 $\mathcal{A}_{\text{II-1}}$ 首先对 ID_s 执行公钥生成询问，当敌手 $\mathcal{A}_{\text{II-1}}$ 对 Γ 发送 $\langle ID_s,ID_r,m\rangle$ 进行签密询问时，Γ 从 L_{pk} 中查询 ID_s 所对应的 $\langle ID_s,X_s,Y_s,c_s\rangle$ 信息，并执行如下操作. 若 $c_s=1$，则 Γ 终止模拟，若 $c_s=0$，则 Γ 对 ID_s 执行私钥生成询问，获取 SK_s 信息，对 ID_r 执行公钥生成询问，并获取 $\langle ID_r,X_r,Y_r,c_r\rangle$ 信息. 运行 SignCrypt 算法，获取密文信息 $\mathfrak{R}=(S,C,h)$，并将密文 $\mathfrak{R}$ 发送至敌手 $\mathcal{A}_{\text{II-1}}$.

解签密询问：敌手 $\mathcal{A}_{\text{II-1}}$ 首先对 ID_r 执行公钥生成询问，获得 $\langle ID_r,X_r,Y_r,c_r\rangle$ 信息，敌手将信息 $\langle ID_s,ID_r,\mathfrak{R}\rangle$ 发送至 Γ 进行解签密询问查询，并根据 c_r 的数值进行如下的计算：如果 $c_r=0$，则 Γ 对 ID_r 执行私钥生成询问，获取 SK_r 信息，对 ID_s 执行公钥生成询问，并获取其公钥信息，然后对密文信息 $\mathfrak{R}$ 执行 UnSignCrypt 算法，并执行校验过程，如果校验通过，返回明文信息 m 给敌手 $\mathcal{A}_{\text{II-1}}$，否则输入的密文信息 $\mathfrak{R}$ 无效，游戏退出. 若 $c_r=1$，当 $\langle ID_s,X_s,Y_s,h_1\rangle\in L_1$，$\langle V_s,h_2\rangle\in L_2$，$\langle ID_s,m,X_s,R_s,h_3^X\rangle\in L_3$，$\langle ID_s,m,Y_s,R_s,h_3^Y\rangle\in L_3$，$\langle ID_s,R_s,C,h_4\rangle\in L_4$，计算 $m\|U=C\oplus h_2$，若 $UG=h_3^X(X_s+Y_s+h_1P_{pub})+h_3^YR_s$ 成立，则返回信息 m 给敌手 $\mathcal{A}_{\text{II-1}}$，否则 Γ 终止模拟，游戏退出.

挑战阶段：

敌手 $\mathcal{A}_{\text{II-1}}$ 选取一对等长的明文信息 m_0 和 m_1，以及两个身份 ID_s 和 ID_r，敌手 $\mathcal{A}_{\text{II-1}}$ 将挑战信息发送至 Γ，Γ 收到敌手发送的明文信息和身份信息后，首先对 ID_r 执行公钥生成询问，Γ 可获得 L_{pk} 中该用户对应身份信息 $\langle ID_r,X_r,Y_r,c_r\rangle$. Γ 生成随机数 $\kappa\in\{0,1\}$，Γ 对消息 m_κ 进行签密. 如果 $c_r=0$，则 Γ 结束，终止游戏，否则得到 x_r^{know} 和 r_r^{know}，令 $R'=bG$，Γ 对 ID_s 执行公钥生成询问，如果 $c_s=1$，得到 x_s^{know} 和 r_s^{know}，否则终止游戏. 计算得到 $y_s^{know}=r_s^{know}+ah_1$，$\Gamma$ 选择满足 $U'G=h_3(X_s+Y_s+h_1P_{pub})+h_3R'$ 的随机数 $U'\in Z_p^*$，选取满足 $V'=\left(x_r^{know}+y_r^{know}\right)R'$ 的 $V'\in Z_p^*$，计算得到 $C'=(m_\kappa\|U')\oplus H_2(V')$ 和 $h'=H_4(ID_s,R',C')$，选取满足 $R'=S'\left(X_s+Y_s+h_1^sP_{pub}+h'G\right)$ 的随机数 $S'\in Z_p^*$，将密文 $\mathfrak{R}=(S',C',h')$ 发送至敌手 $\mathcal{A}_{\text{II-1}}$.

猜测阶段：

敌手 $\mathcal{A}_{\text{II-1}}$ 在 $\kappa'\in\{0,1\}$ 中猜测出一个数值，如果 $\kappa'=\kappa$，则 Γ 可输出 CDH 的解为 $abG=\left(y_s^{know}-r_s^{know}\right)h_1^{-1}R'$，否则没有解决 CDH 问题. 若敌手 $\mathcal{A}_{\text{II-1}}$ 能够以不可忽略优势 ε 成功攻破系统的机密性，而且期间没有终止游戏过程，则 Γ 就可以求解 CDH 问题.

设事件 P 代表敌手 $\mathcal{A}_{\text{II-1}}$ 未询问用户 ID_s 的私钥信息，即 $Pr(P)=\left(1-\dfrac{\xi}{2^\lambda}\right)^2$，事件 P' 表示未终止询问过程，即 $Pr(P')=(1-\delta)^\varsigma$，事件 P'' 表示未终止挑战过程，即 $Pr(P'')=\delta$，则游戏未终止的概率至少为 $Pr(P\wedge P'\wedge P'')=$

$\left(1-\frac{\xi}{2^{\lambda}}\right)^{2}(1-\delta)^{\varsigma}\delta$. 由于 $\delta=\frac{1}{\varsigma+1}$，当 ς 足够大的时候，$(1-\delta)^{\varsigma}\to\frac{1}{e}$，故 $Pr(P\wedge P'\wedge P'')\geqslant\left(1-\frac{\xi}{2^{\lambda}}\right)^{2}\frac{1}{e(\varsigma+1)}$.

综上所述，若 Γ 未终止游戏过程，且敌手 $\mathfrak{R}_{\text{I-1}}$ 能够在多项式时间内以不可忽略优势 ε 赢得相关游戏，则 Γ 能以优势 $Adv^{c}(\mathfrak{R})\geqslant\left(1-\frac{\xi}{2^{\lambda}}\right)^{2}\frac{\varepsilon}{e(\varsigma+1)}$ 解决 CDH 问题.

定理 6. 攻击方式 $\mathfrak{R}_{\text{I}}$ 下本方案能够保证系统的不可伪造性，即敌手 $\mathfrak{R}_{\text{I-2}}$ 不能够通过多项式时间的计算以不可忽略优势 $Adv^{u}(\mathfrak{R})$ 赢得本次游戏.

证明. 同定理 1 证明类似，由于公钥信息存储在区块链中，区块链具有不可篡改的特性，敌手 $\mathfrak{R}_{\text{I-2}}$ 并不能够篡改公钥来发起对系统不可伪造性攻击，因此对于攻击方式 $\mathfrak{R}_{\text{I}}$ 而言，系统方案具有不可伪造性.

定理 7. 攻击方式 $\mathfrak{R}_{\text{II}}$ 下本方案能够保证系统的不可伪造性. 即敌手 $\mathfrak{R}_{\text{II-2}}$ 如果最多能够执行 ς 次签密查询和 ξ 次私钥生成查询操作，就能够在多项式时间内以不可忽略优势 ε 赢得相关游戏，那么系统就能以不可忽略优势 $Adv^{u}(\mathfrak{R})\geqslant\left(1-\frac{\xi}{2^{\lambda}}\right)\frac{\varepsilon}{e(\varsigma+1)}$ 在多项式时间内解决 DL 问题.

证明. 令算法 Γ 为 DL 问题的解决算法，设定输入为 $\langle G,bG\rangle$，其中 $b\in Z_p^*$，而且其数值未知，算法的目标就是求解 b. 算法 Γ 利用敌手 $\mathfrak{R}_{\text{II-2}}$ 攻击系统过程，来求解上述问题. 首先 Γ 执行 Setup 步骤初始化系统环境，得到参数 $Params_{pub}=\langle p,G_p,E,G,q,H_1,H_2,H_3,H_4,P_{pub},\oplus,\|\rangle$，其中 $P_{pub}=bG$，并将 $Params_{pub}$ 发送至敌手 $\mathfrak{R}_{\text{II-2}}$，且系统主密钥 b 不被敌手 $\mathfrak{R}_{\text{II-1}}$ 所知. 维护列表 L_1、L_2、L_3、L_4、L_{sk} 和 L_{pk}，分别用于存储敌手 $\mathfrak{R}_{\text{II-2}}$ 对预言机 H_1、H_2、H_3、H_4 以及私钥生成查询和公钥生成查询的结果，开始时将列表初始化为空.

询问阶段：

按照定理 2 中的询问方式，敌手 $\mathfrak{R}_{\text{II-2}}$ 对随机预言机 H_2、H_3、H_4、H_1 以及私钥生成和公钥生成进行询问.

签密询问：设发送端为 ID_s，接收端为 ID_r，敌手 $\mathfrak{R}_{\text{II-2}}$ 首先对 ID_s 执行公钥生成询问，当敌手 $\mathfrak{R}_{\text{II-2}}$ 对 Γ 发送 $\langle ID_s,m\rangle$ 进行签密询问时，Γ 从 L_{pk} 中查询 ID_s 所对应 $\langle ID_s,X_s,Y_s,c_s\rangle$ 信息，并执行如下操作. 若 $c_s=1$，则 Γ 终止模拟；若 $c_s=0$，则 Γ 对 ID_s 询问其对应的私钥信息 $\langle x_s,y_s\rangle$，对 ID_r 执行公钥生成询问，得到对应的公钥信息，然后运行 SignCrypt 算法，获取密文信息 $\mathfrak{R}=(S,C,h)$，并将密文 $\mathfrak{R}$ 发送至敌手 $\mathfrak{R}_{\text{II-2}}$.

签名校验询问：敌手 $\mathfrak{R}_{\text{II-2}}$ 首先对 ID_s 执行公钥生成询问，并将密文信息 $\mathfrak{R}$ 发送至 Γ 进行签名校验询问，Γ 从列表 L_{pk} 中查询得到 $\langle ID_s,X_s,Y_s,c_s\rangle$，并根据 c_s 的数值进行如下的计算. 如果 $c_s=0$，对 ID_r 执行公钥生成询问，并获取 $\langle ID_r,X_r,Y_r\rangle$ 信息，然后对密文信息 $\mathfrak{R}$ 执行 UnSignCrypt 算法，如果最后校验通过，返回 m 给敌手 $\mathfrak{R}_{\text{II-2}}$，否则终止模拟. 若 $c_s=1$，$\langle ID_s,X_s,Y_s,h_1\rangle\in L_1$，$\langle V_s,h_2\rangle\in L_2$，$\left\langle ID_s,m,X_s,R_s,h_3^X\right\rangle\in L_3$，$\left\langle ID_s,m,Y_s,R_s,h_3^Y\right\rangle\in L_3$，$\langle ID_s,R_s,C,h_4\rangle\in L_4$，计算 $m\|U=C\oplus h_2$，若 $UG=h_3^X(X_s+Y_s+h_1P_{pub})+h_3^YR_s$ 成立，则返回 m 给敌手 $\mathfrak{R}_{\text{II-2}}$，否则 Γ 终止该游戏.

伪造阶段：

敌手 $\mathfrak{R}_{\text{II-2}}$ 选择随机整数 α，α 小于 p，$R^*=\alpha G$，对 ID_s 执行公钥生成询问，如果 $c_s=1$，得到 x_s^{know} 和 r_s^{know}，且满足 $X_s=x_s^{know}G$，$Y_s=r_s^{know}G$，否则终止游戏. 计算得到 $y_s^{know}=r_s^{know}+ah_1$，选择满足 $U'G=h_3(X_s+Y_s+h_1P_{pub})+h_3R^*$ 的随机数 $U'\in Z_p^*$，选取满足 $V'=\left(x_s^{know}+y_s^{know}\right)R^*$ 的随机数 $V'\in Z_p^*$，计算 $C'=(m\|U')\oplus H_2(V')$ 和 $h'=H_4(ID_s,R^*,C')$，选取满足 $R^*=S'\left(X_s+Y_s+h_1^sP_{pub}+h'G\right)$ 的随机数 $S'\in Z_p^*$. 敌手 $\mathfrak{R}_{\text{II-2}}$ 输出身份 ID_s 和伪造密文 $\mathfrak{R}=(S',C',h')$，若敌手 $\mathfrak{R}_{\text{II-2}}$ 伪造成功，并且在 L_{pk} 中的信息满足 $c_s=1$，则 Γ 输出 $b=\left(S^*h_1^s\right)^{-1}\left[\alpha-S^*\left(x_s^{know}+r_s^{know}\right)\right]$，其中 $S^*=\alpha\left(x_s^{know}+r_s^{know}+bh_1^s\right)^{-1}$，即 b 为 DL 问题的答案.

设事件 P 表示敌手 $\mathfrak{R}_{\text{II-2}}$ 未对 ID_s 执行私钥生成询问操作，则 $Pr(P)=1-\frac{\xi}{2^{\lambda}}$，事件 P' 表示未终止询问操作，即 $Pr(P')=(1-\delta)^{\varsigma}$，事件 P'' 表示敌手 $\mathfrak{R}_{\text{II-2}}$ 伪造签名后 Γ 未终止，即 $Pr(P'')=\delta$，则游戏未终止的概率至少为 $Pr(P\wedge P'\wedge P'')=\left(1-\frac{\xi}{2^{\lambda}}\right)(1-\delta)^{\varsigma}\delta$. 由于 $\delta=\frac{1}{\varsigma+1}$，当 ς 足够大的时候，$(1-\delta)^{\varsigma}\to\frac{1}{e}$，故 $Pr(P\wedge P'\wedge P'')\geqslant\left(1-\frac{\xi}{2^{\lambda}}\right)\frac{1}{e(\varsigma+1)}$.

综上所述，若敌手 $\mathfrak{R}_{\text{II-2}}$ 能在多项式时间内以不可忽略优势 ε 赢得相关游戏，且 Γ 没有终止游戏过程，则 Γ 能以优势 $Adv^{u}(\mathfrak{R})\geqslant\left(1-\frac{\xi}{2^{\lambda}}\right)\frac{\varepsilon}{e(\varsigma+1)}$ 解决 DL 问题.

WANG Li-Peng, Ph. D. candidate. His research interests include cryptography and blockchain.

CHEN Zhong, Ph. D., professor. His research interests include cryptography and blockchain.

GUAN Zhi, Ph. D., associate professor. His research interest is cryptography.

LI Qing-Shan, Ph. D. candidate. His research interest is blockchain.

Background

Users can utilize terminal devices such as mobile phones or smart bracelets to collect data, but those data are vulnerable to eavesdropping and tampering. In order to guarantee security and certification, we propose a certificate-less aging signcryption scheme based on the elliptic curve. The proposed method owns the following security attributes: confidentiality, unforgeability, public verification, anonymity, non-repudiation. Based on the discrete logarithm and CDH problems, we prove the new scheme in the random oracle model. Compared with latest related works, the proposed scheme has shorter key length and higher efficiency, which can be adapted to the IoT scenarios.

For precise control of sensing devices, we propose a public key aging mechanism for the proposed signcryption scheme. The new method is based on blockchain to ensure that the devices should exit according to system configurations. Public keys are stored in blockchain which can not be tampered with, and are validated by smart contracts without human participation to ensure credibility. Performance analysis shows that the scheme has high efficiency.

This work was supported by National Key Research and Development Program of China (2020YFB1005404, 2018YFB0803601), Science and Technology Program of Henan Province (202102210359), Henan Province Higher Education Key Research Project (22A520048, 20B520040). Those programs aim to realize a security network model based on blockchain. Currently, we have done some work on them. The paper proposes a public key aging mechanism to precisely control IoT devices, and aims to guarantee security and certification during data transmission.