计算机研究与发展
Journal of Computer Research and Development
DOI: 10.7544/issn1000-1239.202440351
61(10): 2501−2513, 2024

基于分布式数字身份的可审计医疗数据共享

宋文鹏[1,2] 张 亮[1,2,3] 马宇航[3]
[1](复旦大学计算机科学技术学院 上海 200433)
[2](上海区块链工程技术研究中心 上海 200433)
[3](海南大学网络空间安全学院(密码学院) 海口 570228)
(18110240060@fudan.edu.cn)

Auditable Healthcare Data Sharing Based on Decentralized Identity

Song Wenpeng[1,2], Zhang Liang[1,2,3], and Ma Yuhang[3]
[1](*School of Computer Science*, *Fudan University*, *Shanghai* 200433)
[2](*Shanghai Engineering Research Center of Blockchain*, *Shanghai* 200433)
[3](*School of Cyberspace Security* (*School of Cryptology*), *Hainan University*, *Haikou* 570228)

Abstract The continuous rapid development of Internet technology has brought increasing convenience for data sharing. However, data security and privacy issues have also emerged. Taking healthcare as an example, people hope to seek treatment quickly and claim reimbursement promptly, but do not want to disclose their medical records and expenses in hospitals. To adapt to the above data sharing scenarios, we propose a decentralized identity (DID) model based on blockchain and proxy re-encryption (PRE). Further, we achieve a data sharing framework for smart healthcare. Blockchain is leveraged to avoid the problem of single-node failure, ensuring data availability and data consistency. PRE is employed to achieve the separation of data generation, management and usage. This character is identical to that of the DID model, where verifiable credential (VC) issuance and verification are separated. Besides, PRE helps to achieve the goal of patient-centered healthcare. Furthermore, we obtain verifiable presentation (VP) by using the BLS aggregation signatures, which enable us to combine and authenticate multiple VCs. We incorporate blockchain as a decentralized trusted third party to check the correctness of cryptographic operations, achieving data verifiability and audibility in an encrypted mode. Also, we make a scrupulous security analysis of the proposed framework and compare it with related work. Finally, we conduct comprehensive experiments based on Ethereum and IPFS (inter-planetary file system), demonstrating the feasibility and efficiency of our solution.

Key words proxy re-encryption; data sharing; blockchain; smart healthcare; decentralized identity

摘 要 互联网技术的不断快速发展,为数据共享带来日益便利之处,但数据安全和隐私问题也接踵而至.以医疗保健为例,人们既希望能迅速就诊和及时报销,又不希望泄露个人的病历和费用信息.为了适配此类数据共享场景,方案提出一种基于区块链和代理重加密(proxy re-encryption, PRE)的分布式数字身份(decentralized identity, DID)模型,进而实现智慧医疗的数据共享框架.区块链技术能避免单节点故障问

收稿日期: 2024-05-27; **修回日期**: 2024-07-14
基金项目: 国家重点研发计划项目(2019YFB2101703); 国家自然科学基金项目(62272107, U19A2066); 上海市科技创新行动计划(21511102200); 国家自然科学基金青年科学基金项目(62302129); 海南省重点研发项目(ZDYF2024GXJS030)
This work was supported by the National Key Research and Development Program of China (2019YFB2101703), the National Natural Science Foundation of China (62272107, U19A2066), the Innovation Action Plan of Shanghai Science and Technology (21511102200), the National Natural Science Foundation of China for Young Scientists (62302129), and the Hainan Provincial Key Research and Development Program (ZDYF 2024GXJS030).

题,以确保数据的可用性和一致性.代理重加密能够实现数据的生成、管理和使用权限分离,实现 DID 模式下的可验证身份发行和验证,并达到医疗场景中以病患为中心的目的.运用 BLS 签名方案,对该 DID 模式下的身份可验证表达进行聚合签名及认证以提升效率.针对提出的代理重加密方案,在区块链上验证了密文的正确性,实现数据在密态下的可验证性和审计性.最后,方案对所提出的智慧医疗场景进行了安全分析,与相关工作对比分析,并在以太坊和星际文件系统(inter-planetary file system, IPFS)区块链平台上进行系统的实验验证,体现方案的可行性和高效性.

关键词 代理重加密;数据共享;区块链;智慧医疗;分布式数字身份

中图法分类号 TP391

随着国际上《通用数据保护条例》(General Data Protection Regulation, GDPR)和国内《数据安全法》的颁布,人们对于数据隐私泄露越来越担忧,同时也对隐私保护相关技术高度重视.一方面,在新一代互联网医疗场景中,病人的就医记录、诊断信息、治疗方案和医保使用情况等均是居民日益关注的隐私数据[1].另一方面,用户对于自身医疗信息系统中隐私数据的控制和便捷授权要求越来越高,例如在商业保险直接赔付场景,以及在保证最小限度暴露隐私数据的前提下,能够进行自动化商业保险理赔.因此,如何在充分保护隐私的前提下,细粒度、安全可靠地进行医疗信息授权共享,是亟需解决的现实问题.

数字身份作为表示个人或实体在线身份的一种手段,已经成为数字时代中解决数据安全问题的重要技术方案.分布式数字身份(decentralized identity, DID)也称为自我主权身份(self-sovereign identity, SSI),是一项可以辅助解决数据权属和数据流转问题的重要思路[2].区别于传统数字身份技术,DID 以用户为中心,由用户自主控制其数字身份的使用.区块链技术和分布式账本技术的发展,为 DID 构建提供了一种可靠的技术基础设施[3].

W3C 在 2022 年发布 DID 核心架构数据模型[4].图 1 描绘基于区块链的 DID 框架.该图指出 DID 体系至少包括 3 类角色:身份发行方、身份持有方、身份验证方.这 3 类角色围绕可验证凭证(verifiable credential, VC)构成一个有机整体.发行方生成能够标识用户身份的数据,并生成该数据的签名,用于证明其权威性;持有方能够通过查看凭证,并声明对数据和凭证的拥有权;验证方在收到持有方的凭证出示请求后,可以对凭证进行校验,从而使得持有方完成身份或行为证明.DID 核心问题是可验证凭证的构建和使用.

大量的 DID 方案基于区块链技术或者分布式账本技术实现.为避免中心化服务器滥用个人账户信息的风险,Szalachowski[5] 提出一种基于口令认证的 DID 框架,然而,该框架要求区块链平台具有“机密性智能合约”的能力,因此,该研究还使用 Intel SGX[6] 作为可信执行环境提供机密性的硬件保障.Fathalla 等人[7] 提出了基于区块链的可信自治身份管理系统,区块链用于完整性校验.同时,使用主动秘密分享协议[8] 实现 DID 的更新,为不影响区块链完整性校验,在分片的更新过程中采用变色龙哈希算法[9].

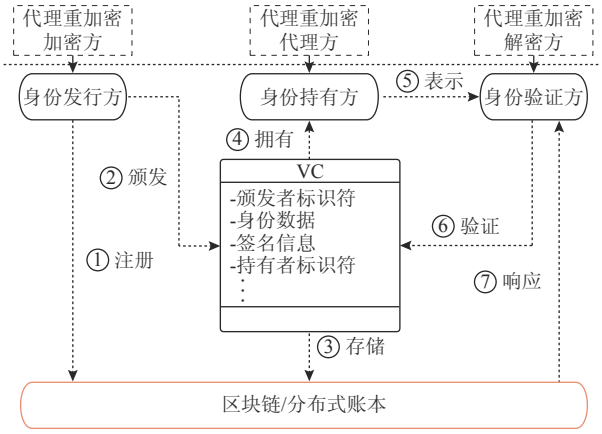


Fig. 1 The framework of decentralized identity

图 1 分布式数字身份框架

医疗场景具有一定特殊性,一方面病患需要听从医嘱,医生是医疗数据的生产者,另一方面,病患的个人隐私包含于医疗数据中,病患又需要对数据拥有控制权和知情权.Zhang 等人[10] 使用属性基密码[11] 构建了以病患为中心的医疗数据共享方案,方案中联合以太坊(Ethereum)[12] 区块链基础协议和 IPFS[13] 作为医疗数据的存储介质.Thilakanathan 等人[14] 使用一种基于 ElGamal 体制的代理重加密[15] 方案,使得病患可以实时并隐私地向医护人员共享健康数据.Zhou 等人[16] 则将基于身份的代理重加密体制应用在无线体联网中,该方案防止代理通过监听不安全信道对数据进行篡改,还重点解决了存在于医疗数据存储在云服务器中的后门渗透攻击问题.Sharma 等人[17] 设计了一种透明的保险索赔流程,并使用智能合约对医疗信息访问进行可审计的跟踪,方案使用

零知识证明验证其身份，由代理重新加密授权病患访问医疗数据. 此外，代理重加密还可以应用在版权保护[18]、物联网[19]、云计算[20]等领域，使得数据的产生、授权和使用可以按实际情况分配.

为保护医疗数据的隐私性，并综合考虑智慧医疗中多角色的权限问题和数据流转问题，本文首次提出基于区块链和代理重加密的 DID 解决方案，并应用在商业保险自动理赔场景中. 在应用代理重加密过程中，医院是医疗数据的生产者，也是数据加密方，而病患是代理方，保险公司是数据使用方. 因此，体现病患才具有医疗数据的拥有权和控制权，极好地满足了社会对数据隐私保护的需求. 在此应用场景中，病患具有向保险公司提出理赔的诉求，并希望有限地向保险公司暴露其医疗数据. 此时，医院是病患身份的发行方，病患是身份的持有方，而保险公司是验证方，保险公司要能确信用于理赔的医疗数据的真实性和权威性. 通常，数据持有方不需要出示所有的 VC 内容，而希望选择性地披露某些信息. 因此，对 VC 进行细粒度选择性排列组合至关重要，本文采用 BLS 聚合签名体制[21]实现 W3C 提及的可验证表达(verifiable presentation, VP)方案. 在数据流转过程中，分别使用以太坊和 IPFS 作为数据登记入口和数据仓库，满足对任意数据的存储需求.

本文的贡献概括为 4 点：

1)首次提出基于区块链和代理重加密的 DID 方案，简化了代理重加密的方案构造，构建最小粒度暴露数据内容的安全共享机制，并应用在以病患为中心的智慧医疗场景中. 将医疗数据的 VC 进行多种维度的自由组合，实现灵活的 VP 生成.

2)通过使用区块链作为通信和验证服务平台，确保医疗数据在密态共享下的一致性和正确性，避免应用场景中实体作弊的可能性. 通过使用门限 BLS 签名体制实现一次性对多个凭证(或断言)的链上验证，降低了验证开销.

3)采用以太坊和 IPFS 作为区块链实验平台，链上链下同步实现 Secpk256k1 椭圆曲线群的代理重加密相关算法，充分测试链下运算的时间开销和链上验证 gas 开销，体现出本文方案的高效性和实用性.

4)分析简化后的代理重加密体制的安全性质，并对比相关代理重加密密码体制，体现本文方案在计算开销上的优势. 此外，本文给出基于该代理重加密体制实现的智慧医疗与其他方案的对比分析，本文方案具有优越性.

1 背景和相关知识介绍

1.1 双线性配对

双线性配对(bilinear pairing)是密码学中的一个概念，它是一种特殊的配对，可以用于安全通信、身份基密码、聚合签名等应用中. 双线性配对有 2 个主要特性：双线性性和非退化性. 双线性指的是对于给定的 2 个元素，无论它们来自哪个群，其配对结果与它们的顺序无关. 非退化性指的是任何 2 个不同的元素进行配对的结果都不等于 1_{GT}.

假设 G_1, G_2 和 G_T 是双线性群，阶为素数 p, e: $G_1 \times G_2 \to G_T$ 是配对操作，上述性质用数学语言描述为：

1)双线性. 对任意的 $u \in G_1, v \in G_2$ 及 $a,b \in \mathbb{Z}_p$，都有 $e\left(u^a, v^b\right) = e(u,v)^{ab}$.

2)非退化性. $e(g_1, g_2) \neq 1_{GT}$，其中 $g_1 \in G_1, g_2 \in G_2$.

1.2 BLS 签名

BLS 签名是一种公钥签名方案，由 Boneh, Lynn 和 Shacham 三位学者提出[22]. BLS 签名具有可聚合的特点，可显著提升对多个消息验签的效率.

BLS 签名包含 6 个算法：

1)初始化算法 $(G_1, G_2, G_T, p, g_2, e, H) \leftarrow BLSSetup(1^{\lambda})$. 输入是安全参数 λ，输出中 G_1, G_2 和 G_T 是阶为 p 的群，g_2 为 G_2 的生成元，双线性对 e: $G_1 \times G_2 \to G_T$，哈希函数 $H: \{0,1\}^* \to G_1$.

2)密钥对生成算法 $(vpk_A, vsk_A) \leftarrow BLS\,KeyGen()$. 随机选择 $vsk_A \leftarrow \mathbb{Z}_q, vpk_A = g_2^{vsk_A}$.

3)生成签名算法 $\sigma \leftarrow Sign(vsk_A, m)$. 输出 $\sigma = H(m)^{vsk_A}$.

4)验证签名算法 $Verify(vpk_A, m, \sigma)$. 如果 $vpk_A \in G_2$, $\sigma \in G_1$, 如果 $e(H(m), vpk_A) = e(\sigma, g_2)$ 则返回“真”，否则返回“假”.

5)生成聚合签名算法 $\sigma \leftarrow AggregateSign((m_1,\sigma_1), \cdots, (m_n,\sigma_n))$. 设每个消息 m_i 对应的签名为 $\sigma_i \leftarrow Sign(vsk_i, m_i)$, 聚合签名输出 $\sigma \leftarrow \prod_{j=1}^{n} \sigma_j$.

6)验证聚合签名算法 $AggregateVerify(\{(m_1, vpk_1), \cdots, (m_K, vpk_K)\}, \sigma)$. 计算 $h_j = H(m_j)$, 如果 $e(\sigma, g_2) = \prod_{j=1}^{K} e(h_j, vpk_j)$ 则返回“真”，否则返回“假”.

1.3 以太坊和星际文件系统

区块链是一种分布式账本，它以块链式数据结构来验证和存储数据，利用分布式节点共识算法来

生产和更新数据,利用密码学方式保证数据传输和访问的安全,并利用由自动化脚本代码组成的智能合约来操作数据.区块链的特性包括多中心化、自动化、高度安全性和可信任性.从基础设施的角度来看,区块链成为了“世界性计算机”,提供不可篡改的存储、数据共享平台和公开可验证的计算引擎.其中,以太坊[12]是一种广泛应用的区块链基础协议,它提供图灵完备的虚拟机引擎,能够执行丰富的智能合约,从而构建去中心化应用.为了解决 DDoS 攻击问题,以太坊引入 gas 机制,使得区块链上任意的存储和计算操作都要消耗一定量的 gas 费用.由于在医疗场景中的应用,我们通过构建以太坊许可链网络来实现本文提出的方案.

由于以太坊的设计是面向非许可链的解决方案,其核心是要保证较高的交易速度,为实现较高的吞吐率、交易及其数据大小有着严格的限制.通常较大的数据,比如多媒体文件等都不适合通过以太坊直接存储,而星际文件系统(inter-planetary file system, IPFS)[13]正好可以弥补这一点.IPFS 也是一种分布式账本系统,采用点对点网络将文件进行分布式存储,其内容寻址方式是通过实际文件的索引,而没有采用文件的名称或位置,因此其检索和定位方式也要依赖文件的索引来进行.将 IPFS 与以太坊区块链进行结合,可以解决以较低 gas 开销存储任意大小文件的问题.

1.4 代理重加密

代理重加密(proxy re-encryption, PRE)是一种可以对密文进行安全转换的密码学方法[15,23].在使用代理重加密体制的系统中,通常包含数据生成方A、数据代理方P和数据使用方B等角色.代理重加密体制通常包含如下算法:

1)公私钥生成算法$(pk_i, sk_i) \leftarrow PREKeyGen(1^{\lambda})$(数据生成方$A$和数据使用方$B$执行).输入安全参数$\lambda$,输出$(pk_i, sk_i)$,其中$pk_i$为公钥,$sk_i$为私钥.

2)加密算法$C = (capsule, encData) \leftarrow Encrypt(pk_A, M)$(数据生成方$A$执行).其中输入$pk_A$是数据生成者的公钥,$M$是待加密的数据明文.输出$capsule$是算法执行过程中生成的与对称密钥有关的密文信息,$encData$是实际数据的密文信息.

3)解密算法$M \leftarrow Decrypt'(sk_A, C)$(数据生成方$A$执行).其中输入$sk_A$是数据生成方的私钥,$C$是密文,输出$M$是数据明文信息.

4)密钥重分发算法$kFrag \leftarrow ReKeyGen(sk_A, pk_B)$(数据生成方$A$执行).其中输入$sk_A$是数据生成方$A$的私钥,$pk_B$是数据使用方$B$的公钥,输出$kFrag$是对密文进行重加密的密钥碎片,数据生成方将$kFrag$发送给数据代理方$P$.

5)重加密算法$(cFrag, \pi) \leftarrow ReEnc(kFrag, capsule)$(数据代理方$P$执行).该算法输出的$cFrag$是对$capsule$进一步密码学操作,即使用密钥碎片$kFrag$对$capsule$进行隐私保护,$\pi$是零知识证明,用于证明代理方的诚实性.最后,假设新的密文$C' = (cFrag, encData)$.

6)解密算法$M \leftarrow Decrypt(sk_B, pk_A, C', cFrag)$(数据使用方$B$执行).该算法中数据使用方$B$使用其的私钥$sk_B$和密文$cFrag$对重加密后的密文$C'$进行解密,得到明文数据$M$.

本文借助 UMBRAL 代理重加密体制[23]的思想,对该代理重加密体制简化成单代理,从而构建本文提出的 DID 方案的密码原语.

2 系统模型及安全模型

2.1 系统实体

在本节中,我们将介绍系统中涉及的各个实体,用来描述实体的符号代理重加密方案.

1)患者P.在本系统环境中指凭证持有人,病患在就医过程中会产生用药、住院、支付等行为活动.病患是就医数据的拥有者,具有数据的控制权和委托处理权.

2)医院A.在本系统环境中指凭证发行方,同时,医院还是本系统中的数据生成方.通过医生对病患的接诊和治疗,并如实记录相关过程,最终医院收取一定的医疗费用.

3)银行.在本系统环境中指另一种凭证发行方,如实记录病患的个人身份信息,包括姓名、身份证号、银行卡号和手机号等.

4)保险公司B.在本系统环境中指凭证验证方,通过验证对病患提供的就医活动,实现对购买商业保险的病患进行智能化理赔.

5)断言.某一个事实,通常用于描述凭证持有人的客观情况.该事实可以是凭证持有人的固有属性、被赋予的公认特征或发生在该实体上的事件.常用 JSON 格式表示.

6)VC.由凭证发行方对 1 个或多个断言形成的可验证证明,通过将凭证交给凭证验证方完成验证.

7)以太坊.一方面作为分布式存储实现安全存储,记录需要在医院、银行和保险公司中共享的相关数据索引和数据本身;另一方面作为智能合约的可信计算执行引擎,自动化地对相关实体提交的数据

进行验证.

8)IPFS.实际加密数据的分布式文件存储数据库或数据仓库,解决以太坊只能存储少量数据的问题.

2.2 系统目标

1)隐私性.个人医疗数据应对商业保险公司和银行等非医疗相关机构具有隐私保护性.

2)病患为中心.患者需要对其医疗数据具有拥有权、控制权和知情授权的能力.

3)最小暴露原则.患者可以选择性地以最小限度地披露病历中的诊断信息,用于智能化理赔.

4)可验证性.①系统中的数据生成方应保证身份认证;②VC 的正确性应保证公开可验证.

5)可审计性.数据代理方执行密文重加密的过程需要可审计.

6)自治性.各凭证发行方互相独立,无须建立通讯或沟通协商.

2.3 系统威胁

本系统考虑 4 个多项式时间的敌手$\mathcal{A}$可能形成的潜在安全威胁:

1)敌手$\mathcal{A}$冒充或监听数据生成方(医院),修改医疗数据,生成对应密文,并伪造签名认证信息.

2)敌手$\mathcal{A}$冒充数据代理方(病患),对链上的合法密文进行非法重加密(即伪造rk),从而欺骗凭证认证方或者导致病患无法理赔.

3)敌手$\mathcal{A}$冒充病患,非法请求理赔,试图获得保险公司的赔付.

4)敌手$\mathcal{A}$控制凭证验证方(保险公司),尝试获得与病患理赔项目无关的医疗数据.

3 系统设计

3.1 系统概述

本系统采用区块链和代理重加密实现了 DID 方案,并用于商业保险自动理赔场景,提升医院和保险公司的协同效率.图 2 展现基于区块链和代理重加密的商业保险自动理赔流程图.图中的实体包括:病患 P 作为代理重加密密码体制中的数据代理方;医院 A 作为代理重加密密码体制中的数据持有/生成方;保

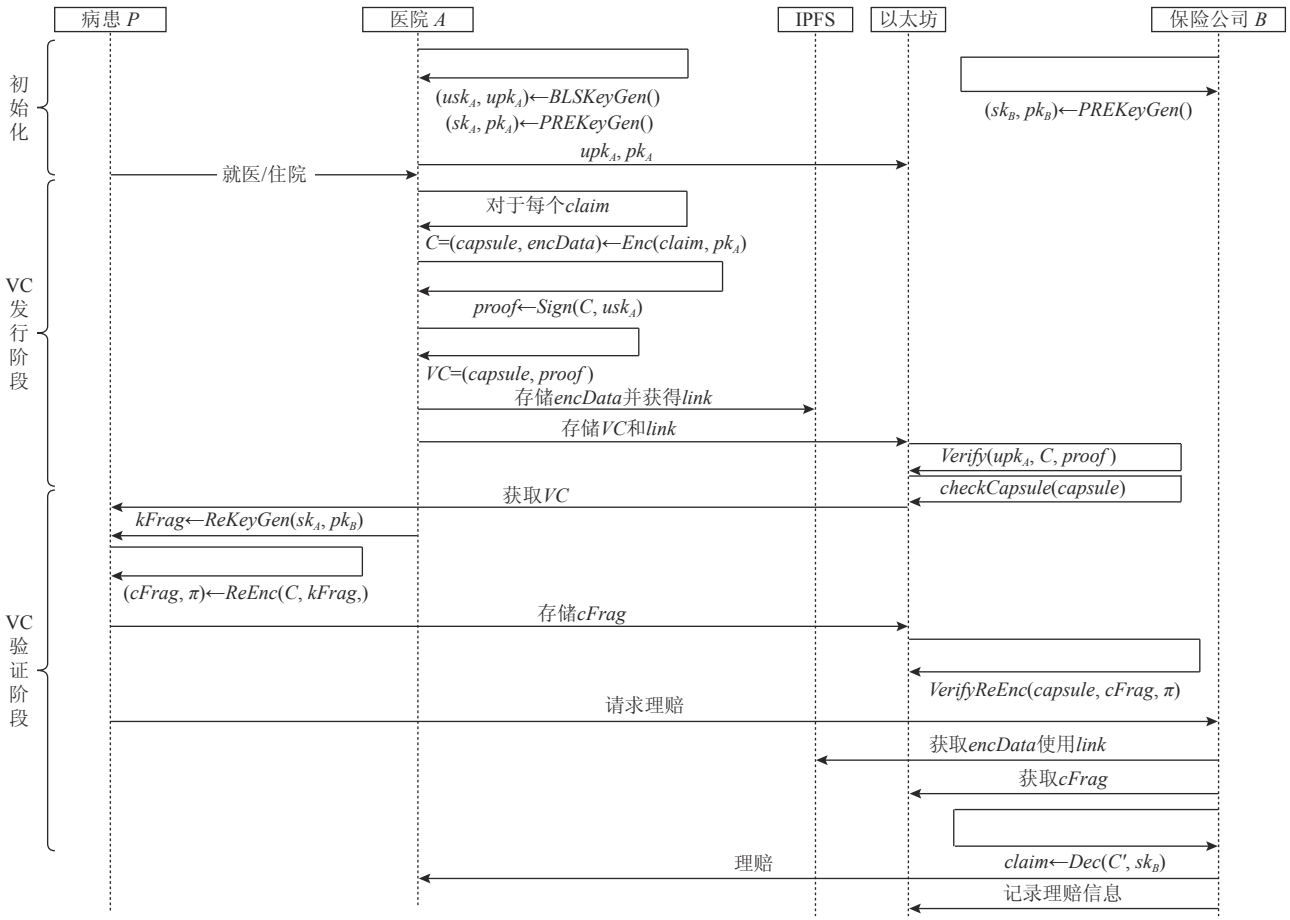


Fig. 2 Blockchain and proxy re-encryption based DID framework and its application in auditable healthcare system

图 2 基于区块链和代理重加密的 DID 框架及其在可审计智慧医疗系统中的应用

险公司 B 作为代理重加密密码体制中的数据验证方; 区块链是可验证凭证的存储方, 也是凭证的自动化验证和校验方, 同时还是代理重加密过程准确性验证的保障方.

流程图从时序上主要分为初始化、凭证发行和凭证验证 3 个阶段. 在初始化阶段, 医院分别初始化 BLS 签名密码体制和代理重加密密码体制, 并将公钥 vpk_A 和 pk_A 上传至区块链网络存储. 在发行阶段, 主要由病患、医院和区块链之间完成数据的生成及交互. 在验证阶段, 主要由病患、保险公司和区块链之间完成数据的认证及交互. 由此可见, 发行阶段和验证阶段中不存在医院和保险公司的交互, 一方面通过降低实体之间的耦合性提升自动理赔的效率, 另一方面通过隔离实体保护非理赔病患用户的个人隐私.

保险公司通过对医院生成的可验证凭证进行验证和对病患提供的重加密数据进行验证实现自动化理赔, 避免病患多跑路和先行垫付医药费. 最后将理赔事实进行存证, 供保险监管机构备用复核和审查.

3.2 初始化

医院 A 作为数据生成方分别执行代理重加密和 BLS 密码体制的密钥生成算法, 保险公司 B 作为数据的使用方也执行代理重加密的密钥生成算法. 具体调用为:

$$(vsk_A, vpk_A) \xleftarrow{R} BLSKeyGen(),$$

$$(sk_A, pk_A) \xleftarrow{R} PREKeyGen(),$$

$$(sk_B, pk_B) \xleftarrow{R} PREKeyGen().$$

3.3 凭证发行

具体来讲, 本文提出的智慧医疗场景中的凭证发行过程为:

1)病患 P 于某医院 A 就医或进行住院治疗, 此过程产生一定的医疗费用, 并且 P 希望使用购买过的商业医疗保险覆盖一部分费用.

2)医院 A 产生一条断言, 医院 A 使用代理重加密的加密算法对上述断言进行加密得到密文 $C = Enc(claim, pk_A)$. 其中 Enc 函数实现为:

$$r, u \xleftarrow{R} \mathbb{Z}_q, E = g^r, V = g^u,$$

$$s = u + r \cdot H_2(E, V),$$

$$K = KDF((pk_A)^{r+u}),$$

$$encData = AESEncrypt(claim, K), capsule = (E, V, s),$$

$$C = (capsule, encData),$$

其中 H_2 是哈希函数, KDF 是密钥扩展算法, $AESEncrypt$ 是 AES 对称加密算法.

3)医院 A 使用其签名私钥 vsk_A 对上述密文 C 进行签名得到签名结果 $proof \leftarrow Sign(vsk_A, C)$.

4)医院 A 构造 $VC = (C, proof)$, 并提交至区块链进行持久化存储. 区块链智能合约自动校验 $capsule$ 的正确性, $CheckCapsule$ 函数的定义为:

$$g^s \stackrel{?}{=} V \cdot E^{H_2(E,V)}.$$

3.4 凭证验证

凭证验证的详细过程为:

1)医院 A 作为代理重加密体制中的数据生成方, 为病患 P 执行密钥重分发算法得到 $kFrag \leftarrow ReKeyGen(sk_A, pk_B)$, 并将 $kFrag$ 通过安全信道发送给病患 P, 同时医院 A 提供对 X_A 字段的签名. 其中 $ReKeyGen$ 函数实现为:

$$x_A \xleftarrow{R} \mathbb{Z}_q, X_A = g^{x_A},$$

$$d = H_3(X_A, pk_B, (pk_B)^{x_A}),$$

$$kFrag = (rk = sk_A/d, X_A),$$

其中 H_3 是哈希算法.

2)病患 P 从区块链获得某断言的密文 $C = (capsule, encData)$, P 作为代理重加密中的数据代理方执行重加密算法得到 $(cFrag, \pi) \leftarrow ReEnc(kFrag, capsule)$, π 是使用 Sigma 协议和 Fiat-Shamir 转换生成密文 C' 对应的非交互零知识证明. 并令新的密文 $C' = (cFrag, encData)$. $ReEnc$ 函数实现为:

$$E_1 \leftarrow E^{rk},$$

$$V_1 \leftarrow V^{rk},$$

$$cFrag = (E_1, V_1, X_A),$$

$$\tau \xleftarrow{R} \mathbb{Z}_q, E_2 \leftarrow E^{\tau}, V_2 \leftarrow V^{\tau},$$

$$h \leftarrow H_1(E_1, E_2, V_1, V_2, X_A),$$

$$\rho \leftarrow \tau + h \cdot rk,$$

$$\pi = (E_2, V_2, h, \rho),$$

其中, $kFrag = (rk, X_A), capsule = (E, V, s)$.

3)病患 P 将 C' 和 π 上传至区块链, 区块链智能合约自动调用 $VerifyReEnc(capsule, cFrag, \pi)$, 用以自动验证 P 重加密的正确性和合法性. 其中 $VerifyReEnc$ 函数实现为:

$$E^{\rho} \stackrel{?}{=} E_2 \cdot E_1^h,$$

$$V^{\rho} \stackrel{?}{=} V_2 \cdot V_1^h.$$

4)病患 P 向保险公司 B 发起理赔请求, B 访问区块链获得密文 C', 并调用代理重加密体制的解密算法得到断言信息 $claim \leftarrow Dec(C', sk_B)$. 其中 Dec 函数

（本文忽略使用A的私钥sk_A解密初始密文C的算法）实现为：

$$d = H_3\left(X_A, pk_B, X_A^{sk_B}\right),$$

$$K = KDF\left((E_1 \cdot V_1)^d\right),$$

$$claim = AESDecrypt(encData, K),$$

其中H_3是哈希算法，$AESDecrypt$是AES对称解密算法.

5）保险公司B通过自动化解析断言JSON信息，通过判断该断言是否在其保单中，从而帮助病患完成实时理赔. 最后，保险公司将理赔信息存证在区块链网络中.

4 聚合验证

4.1 凭证聚合验证

从图2可以知道，医院作为发行方，提供凭证为病患实现自动化理赔. 在某些特殊断言存在的时候，由于保险公司需要对断言进行人工复核，因此不能实时理赔. 故存在病患垫付医疗费，并提供银行卡号给保险公司以便在就医后进行赔付的情况. 此时，需要医院和银行分别作为独立的发行方，一个作为就医事实断言的发行方，一个作为身份事实断言的发行方. 甚至还有些情况下，病患于多家医院进行了就医，最终需要合并进行理赔，而不同的医院也构成不同的发行方. 此时，对多个VC的聚合管理是VP的一种体现.

为原子化地处理这些多发行方情况，进行凭证层面的动态组合生成VP，可以更加灵活和便捷地使用凭证，也能通过一次运算实现对多条凭证的验证. 下文抽象了存在多个发行方并实现签名聚合的方法.

凭证发行方生成凭证时，根据每个断言密文C_i的证明$proof_i \leftarrow Sign(vsk_i, C_i) = H(C_i)^{vsk_i}$，其中$vsk_i$是第$i$个发行方的私钥，$n$个断言密文签名的聚合算法$AggregateSign((C_i, proof_i), \cdots, (C_n, proof_n))$结果为：$proof \leftarrow \prod_{i=1}^{n} proof_i$. 在区块链上验证各密文聚合签名正确性时，智能合约根据式（1）判断是否成立，成立则完成上链存储.

$$e(proof, g_2) \stackrel{?}{=} \prod_{i=1}^{n} e(proof_i, vpk_i). \tag{1}$$

4.2 断言聚合验证

4.1节介绍了凭证聚合验证的方法，可以解决不同发行方断言密文的聚合问题. 因为某一病患就医可能有多项药物开销或者多次住院，在这些场景中，病患希望将多个聚合产生的费用进行统一理赔. 因此，另一种更为迫切的需求是对同一发行方的多个断言密文进行签名聚合. 本节提出一种同一发行方多个断言密文聚合的方法，该方法不仅能批量验证签名，还能使得发行方对多个断言仅做1次加密操作，大幅度提升效率.

医院A产生若干条断言$\{claim_1, \cdots, claim_n\}$，医院$A$使用代理重加密的加密算法对上述断言进行加密得到密文$C = Enc(claim_1 \| \cdots \| claim_n, pk_A)$. 然后，$A$用其签名私钥$vsk_A$对上述密文进行签名得到结果$proof$. 最后，$VC = (C, proof)$，并提交至区块链网络.

5 实验结果及分析

为验证本文提出方案的正确性和效率，本文实现各算法，并进行系统测试. 实验在VMware Workstation 16的Ubuntu 22.04.2 LTS虚拟机上进行，Intel® Core™ i5-8300HCPU @ 2.30 GHz ×2，4核处理器，4 GB RAM.实验的区块链平台为Ganache的以太坊区块链网络.

代理重加密是构建本文提出的DID的核心密码原语，同时对称加密也不可避免地需要用于加密医疗数据. 图3显示随着断言数据尺寸的增大，对称加密部分所需要的时间开销随之增加，在断言尺寸为1 000 kb时，加解密时间在4 ms左右.

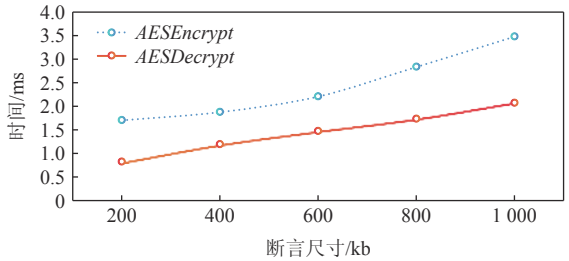


Fig. 3 Time cost of AES symmetric encryption and decryption algorithm

图3 AES对称加解密算法时间开销

图4展示随着断言尺寸变化的代理重加密（即$Enc, ReEnc, ReKeyEnc, Dec$）的时间开销变化图. 由于$Enc$和$Dec$分别封装$AESEncrypt$和$AESDecrypt$函数，因此，它们的时间开销随着断言尺寸线性增长. 而$ReEnc$和$ReKeyEnc$的时间开销几乎保持不变，因为这些算法包含的密码操作是固定的. 具体来说，$ReKeyEnc$的输出（即$kFrag$）包含2个固定元素，rk和X_A.$ReEnc$的输出（即$kFrag$和π）包含7个固定的元素，$E_1, V_1, X_A, E_2, V_2, h, \rho$.

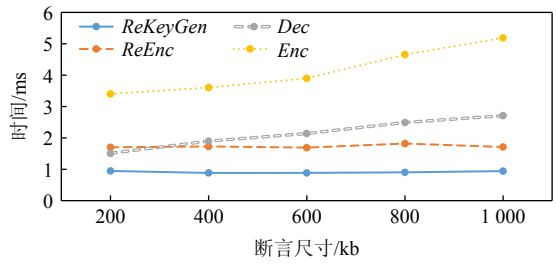


Fig. 4 Time cost of the PRE algorithm

图 4 代理重加密算法时间开销

图 5 描绘签名算法 *Sign* 随着断言尺寸的增加时间开销也随之增加, 以及当固定断言尺寸为 5 时, 聚合签名算法*AggregateSign*随着断言尺寸增加的时间开销. 这 2 个算法的时间开销几乎都保持在常数级别, 这是由于单签名和聚合签名是对密文映射在椭圆曲线的点进行运算, 当明文数据尺寸增加时, 映射到椭圆曲线上的点不变.

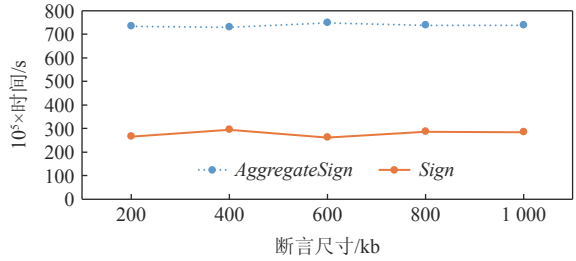


Fig. 5 Signature time cost increases with claim sizes

图 5 签名时间开销随断言尺寸增加

在智能合约中处理 VC 时, 区块链需要开辟存储空间和消耗计算资源. 当医院在上传加密后的医疗数据*encData*到 IPFS 时, 由于 IPFS 索引是一串固定长度的字符, 所以无论医疗数据(断言)的尺寸多大, *encData*的存储开销保持不变.

图 6 细化了链上存储密文时各部分(即*capsule*, *proof*, *cFrag*和*ipfslink*)所需要的 gas 开销. 图 7 展现链上验证所需要的 gas 开销, 包括验证单个断言签名的*Verify*算法、检查*capsule*的*CheckCapsule*算法, 以及校验重加密密文的*VerifyReEnc*算法. 针对尺寸不同的断言, 这些算法处理的数据大小不变, 因此 gas 值消耗不变, 所以断言尺寸不影响验证的效率和开销.

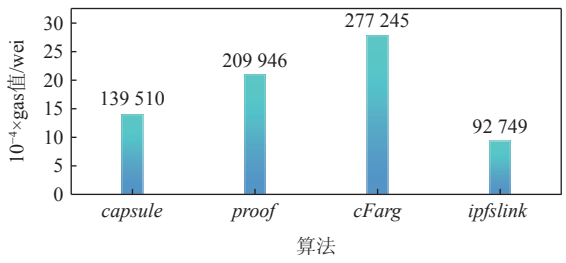


Fig. 6 gas cost of onchain storage

图 6 链上存储的 gas 消耗

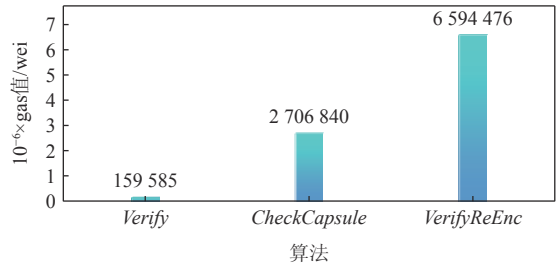


Fig. 7 Single claim verifies gas cost

图 7 单个断言验证 gas 开销

在凭证聚合的场景中, 对多个 VC 进行聚合会产生一定的链下计算开销, 链上有对应的 gas 存储和验证开销. 图 8 显示当断言尺寸固定为 200 kb 时, 随着断言数量的增加聚合签名的时间也随之增加. 这是由于当聚合的签名增多时, 需要处理的椭圆曲线群运算增加, 从而导致聚合签名时间开销增加. 图 9 显示在明文断言尺寸固定为 200 kb 的情况下, 随着断言数量的增加, 上传和验证聚合签名的 gas 值也随之增加. 随着断言数量增加, 验证聚合签名需要的用户公钥和数据映射后的哈希点增加, 因此上传聚合签名及其验证时需要的参数数量增加, 所以消耗的 gas 值增加. 随着断言数量的增加, 验证聚合签名需要的计算量增加, 所以消耗的 gas 值随断言数量增加.

图 10 展示当断言数量固定为 5 的情况下, 随着

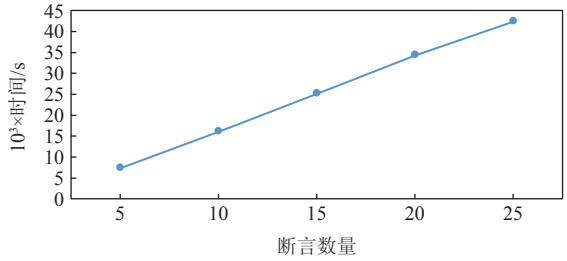


Fig. 8 Time cost of aggregating signatures increases with the number of the claims

图 8 聚合签名时间开销随断言数量增加

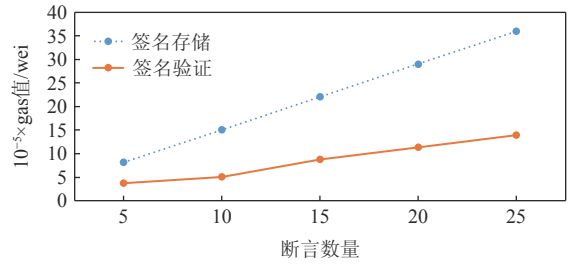


Fig. 9 gas cost stored by aggregated signature increases with the number of claims

图 9 聚合签名存储的 gas 消耗随断言数量增加

断言尺寸的增加，上传和验证聚合签名的 gas 值几乎不变. 这是因为聚合签名是上传密文数据映射的椭圆曲线点签名，与明文尺寸增加导致的密文尺寸增加无关系. 经上述分析可得，在凭证聚合场景下，断言的尺寸不影响聚合验证的开销，断言的数量是影响凭证聚合开销的关键.

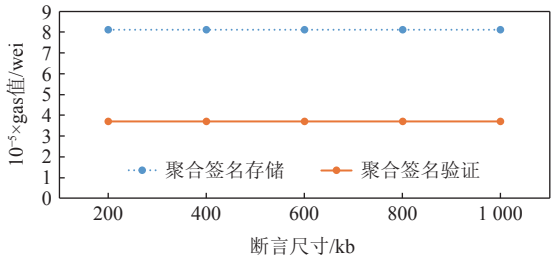


Fig. 10 gas cost stored by aggregated signature varies with claim sizes

图 10 聚合签名存储的 gas 消耗随断言尺寸变化

6 系统分析

6.1 正确性分析

定理 1（正确性）. 本文提出的代理重加密过程是正确的，即若对于任意的断言字符串 $claim$，其密文 $C \leftarrow Enc(claim, pk_A)$ 对应的重加密密钥 $kFrag \leftarrow ReKeyGen(sk_A, pk_B)$ 和重加密密文 $(cFrag, \pi) \leftarrow ReEnc(kFrag, capsule)$，均有 $claim \leftarrow Dec(C', sk_B)$.

证明. 首先，Dec 算法中计算 d 的过程本质是首先数据生产方（医院 A）和数据使用方（保险公司 B）完成一次 DH 交换[24]，即 $pk_B^{x_A} = X_A^{sk_B}$ 恒成立. 其次，由于 $E_1 = E^{rk}, V_1 = V^{rk}$，且 $rk = sk_A/d$，从而 $(E_1 \cdot V_1)^d = (E \cdot V)^{sk_A} = g^{(r+u)sk_A} = pk_A^{r+u}$，即可以恢复出用于对称加密的密钥 K. 最后，调用 $AESDecrypt$ 即可恢复出明文 $claim$.

证毕.

定理 2（$capsule$ 可验证性）. 任何第三方可以使用 $CheckCapsule$ 算法判断数据生产方生成的 $capusule$ 的合法性.

证明. 对 $CheckCapsule$ 算法中的右式进行推导：

$$V \cdot E^{H_2(E,V)} = g^u \cdot (g^r)^{H_2(E,V)} = g^{u+r \cdot H_2(E,V)} = g^s.$$

由代理重加密体制可以看出，重加密的过程就是对 $capsule$ 的重加工处理，$capsule$ 的合法性是重加密函数 $ReEnc$ 能够被正确执行的必要条件. 因此，对 $capsule$ 进行合法性检测是避免加密者滥用数据的重要一环.

证毕.

定理 3（重加密可审计性）. 任何第三方可以判断定理 1 中代理方的诚实性，即 $VerifyReEnc$ 能使用非交互的零知识证明 π 判断 $cFrag$ 的有效性.

证明. 分别对 $VerifyReEnc$ 算法中的右式进行推导：

$$E_2 \cdot E_1^h = E^\tau \cdot (E^{rk})^h = E^{\tau + rk \cdot h} = E^\rho,$$
$$V_2 \cdot V_1^h = V^\tau \cdot (V^{rk})^h = V^{\tau + rk \cdot h} = V^\rho.$$

证毕.

由此可见，$VerifyReEnc$ 以 $capsule$，$cFrag$，π 为输入，使用 Sigma 协议产生零知识证明数据 π，从而有效地判断重加密密文 C' 中 $cFrag$ 的正确性. 同时，本文还要求医院提供的重加密密钥 X_A 字段的签名，用以确保重加密代理方身份的真实性，防止重加密功能的非法滥用.

定理 4（安全性）. 如果密钥扩展算法 KDF、哈希函数（即 H_1，H_2，H_3）、对称加密算法 AES 是安全的，并且离散对数问题是困难的，则本文提出的代理重加密过程是安全的，即密文 C 和重加密密文 C' 均不会泄露明文 $claim$ 的任何有效信息.

证明. 首先，密文 C 和重加密密文 C' 共享数据的 AES 对称加密密文部分 $encData$，如果 AES 算法是安全的，那么接下来需要讨论密钥 K 的安全性. 对称密钥是 $K = KDF((pk_A)^{r+u})$，它的安全性依赖密钥扩展算法 KDF 和输入数据 $(pk_A)^{r+u}$. 如果 KDF 是安全的，那么就只需要考虑输入数据的隐私性. 由于 r 和 u 均为数据生成者随机挑选的值，故 $r+u$ 和 $(pk_A)^{r+u}$ 在其值空间内也具有随机性和均匀分布性，即保证对称密钥 K 的安全性. 其次，密文 C 包含 $capsule$ 字段，$capsule = (E, V, s)$，其中 E 和 V 是以 r 和 u 为离散对数的群元素，当离散对数困难问题成立，由 E 和 V 不能得出 r 和 u 的任何有效信息，从而也不会泄露任何有关密钥 K 的信息；而 $s = u + r \cdot H_2(E, V)$ 也是均匀分布在 $\mathbb{Z}_q$ 上，不会泄露 r，u，K 相关的任何有效信息. 最后，重加密密文 C' 包含 $(cFrag = (E_1, V_1, X_A), \pi)$，其中 E_1 和 V_1 的安全性来源于重加密密钥 $kFrag$ 中的 rk 字段在 $\mathbb{Z}_q$ 中的均匀分布性. 而 π 是由 Sigma 协议和 Fiat-Shamir 转换得到的非交互零知识 NIZK 证明，由 NIZK 的“零知识性”可以得知，π 不泄露任何有关 rk 的隐私信息.

证毕.

定理 5（断言可聚合性）. 第 4.2 节提出的断言粒度的动态组合方法是正确的.

证明. 该方法仅对明文 $claim$ 进行字面字符串追加处理，因此结论显而易见.

证毕.

定理 6（凭证可聚合性）. 第 4.1 节提出的凭证粒度的动态组合方法的验证过程是正确的.

证明. 对式（1）从左往右的推导为：

$$e(proof,g_2)=e\left(\prod_{i=1}^{n}proof_i,g_2\right)=$$
$$\prod_{i=1}^{n}e(proof_i,g_2)=\prod_{i=1}^{n}e\left((H(C_i))^{vsk_i},g_2\right)=$$
$$\prod_{i=1}^{n}e\left(H(C_i),g_2^{vsk_i}\right)=\prod_{i=1}^{n}e(proof_i,vpk_i)$$

证毕.

6.2 系统威胁分析

对2.3节提出的系统威胁进行分析,证明多项式时间敌手$\mathcal{A}$无法在本系统中获得不可忽略的优势.

1)威胁1. 敌手$\mathcal{A}$冒充或监听数据生成方(医院),修改医疗数据,生成对应的密文,并伪造签名认证信息.

可以分析得到:一方面,当医院已经对医疗数据加密得到密文C,定理1已经证明合法的密文C包含对称加密和DH交换,因此,敌手$\mathcal{A}$无法通过延展性攻击对C进行修改,即无法获得合法的密文.另一方面,如果敌手$\mathcal{A}$通过自行对伪造的医疗数据进行加密得到密文C,由于无法对密文C进行生成合法的签名$proof$,那么也就无法通过区块链的自动认证流程,从而其伪造的医疗数据无法成功上链.

2)威胁2. 敌手$\mathcal{A}$冒充数据代理方(病患),对链上的合法密文进行非法重加密(即伪造rk),从而欺骗凭证认证方或者导致病患无法报销.

可以分析得到:首先,从理论上来说,敌手$\mathcal{A}$可以随机选取$kFrag$对任意的密文C进行重加密,即任意选取$rk'\neq rk$,并且成功通过$VerifyReEnc$对重加密过程的验证.为保证仅数据生成方授权过的病患能提交重加密密文,本方案要求医院A在生成重加密密钥$kFrag$后,对其中的X_A字段进行签名,确保X_A可公开的同时,保证了重加密代理方(病患)调用重加密过程的合法性.也就是说,病患上传重加密密文C'时,还会上传X_A及医院对其生成的签名,确保区块链能够验证重加密者身份的真实性.因此,敌手$\mathcal{A}$无法通过伪造$kFrag$对密文进行重加密操作和通过区块链验证,导致病患病患无法报销.

3)威胁3. 敌手$\mathcal{A}$冒充病患,非法请求报销,试图获得保险公司的赔偿.

可以分析得到:敌手$\mathcal{A}$冒充病患时,一方面,如果真实的病患及其病历存在,那么保险公司如实将保险金额打入真实病患的账户;另一方面,如果敌手$\mathcal{A}$构造虚假的病患,当其请求报销时,由于区块链的可追溯性,无法查证该虚假病患的历史数据,因此,保险公司也不会支付任何保险费用.综上,敌手$\mathcal{A}$并不能通过非法请求报销获得任何优势或赔偿金.

4)威胁4. 敌手$\mathcal{A}$控制凭证验证方(保险公司),尝试获得与病患报销项目无关的医疗数据.

可以分析得到:本方案使用代理重加密体制对病患的医疗数据进行加密存储.即使拿到相应密文,敌手$\mathcal{A}$也无法获得其明文信息.同时,本方案的明文对象是断言,即可用于报销的最小粒度医疗数据.当病患提出报销的申请时,才会通过本方案将该报销项目有关的医疗信息共享给保险公司,以便其对报销内容进行核实.如果敌手$\mathcal{A}$控制保险公司,对于病患未申请报销的医疗数据,$\mathcal{A}$由于没有得到对应的合法重加密密文,因此,也无法解密未申请报销的医疗数据.

6.3 性质分析

1)隐私性.本文采用代理重加密算法,由医院(数据生成者)产生并对原始就医记录$claim$进行加密得到密文C,同时,病患使用重加密函数对原始数据进行二次加密处理得到C',由定理4可知,C和C'均不暴露病患就医记录$claim$的任何有用信息,因此本方案具有隐私性.

2)以病患为中心.就医行为具有特殊性,医院是病患数据的生产者,而病患应当是数据的拥有者和实际控制人.本文通过使用代理重加密算法,使得病患作为代理方,成为密态数据的实际处置者,实现了以病患为中心的知情同意、授权管理和报销凭证的有序流转.

3)最小暴露原则.本文通过定义不可分割的“断言”信息,允许医院对多个断言直接进行拼接,生成可验证凭证,也允许病患对来自于多方的可验证凭证进行聚合.定理5和定理6分别证明了对断言和可验证凭证聚合的正确性.因此,可以在实际业务层面设计任意的“断言”信息披露方案,即保证实现最小暴露原则.

4)可验证性.本文一方面要求通过使用BLS签名算法实现数据生成方的不可否认性,另一方面要求密态数据中的$capsule$的可验证性.定理2证明了使用$CheckCapsule$的公开可验证性.同时,BLS签名和$capsule$的正确性验证都由以太坊智能合约自动化完成.

5)可审计性.数据代理方(病患)通过行使对医疗数据的控制权,通过调用$ReEnc$算法实现数据的安全流转.病患的行为可以通过$VerifyReEnc$算法来判断其诚实性和正确性.定理3证明了代理方的可审计性.同时,该审计工作由以太坊智能合约自动化完成.

6)自治性.第4.1节介绍聚合多个数据生产方的

可验证凭证的场景，在产生凭证的过程中，各凭证发行方互相独立，无须建立通讯. 在聚合凭证过程中，通过*AggregateSign*函数输入各断言密文和相应凭证，也无须凭证方进行协商和合作. 因此，凭证发行方之间具有自治性.

6.4 相关工作对比

Thilakanathan 等人[14]仅采用代理重加密被动地保护病患的实时健康数据. Zhou 等人[16]提及的无线体联网方案中，病患可以构建授权信息，因此其具备“以病患为中心”的性质，此外，该方案还使用云服务器作为多个医院沟通的桥梁，并在代理重加密层面做安全保护，因此也具备“自治性”的特点. Zhang 等人[10]以病患为属性密码权威中心，实现“以病患为中心”的特点，也使用属性实现细粒度的就诊信息访问权限，确保了“最小暴露原则”. Sharma 等人[17]使用代理重加密保证医疗记录的隐私性，并让病患使用智能卡确保“以病患为中心”，进一步用 zk-SNARKs 零知识证明[25]验证代理重加密的密钥正确性，实现“可验证性”. 表 1 汇总了相关智慧医疗的性质对比.

众所周知，椭圆曲线指数和双线性配对运算的次数是衡量某一个密码体制性能的重要方式. 在某些平台中，双线性配对耗时可能远比指数运算耗时[26]. 因此，相比较而言，研究人员更倾向构造和使用没有双线性配对的密码体制. 表 2 对比代理重加密体制的性能(表 2 中忽略实际对数据进行对称加密的开销)，表 2 中除 UMBRAL 和本文方案之外，都用到了双线性配对计算 Wang 等人[27]、Agyekum 等人[19]、Sun 等人[28]和 Zhou 等人[16]的方法均基于身份的代理重加密，而基于身份的密码体制通常需要使用双线性配对运算. 从表 2 可以看出 UMBRAL 和本文方案没有使用双线性配对，并且这 2 个的方案支持对*capsule*的检查和验证重加密密文的正确性. 与本文方案验证*capsule*类似，Sun 等人[28]也验证了原始密文中部分字段的正确性. 将*CheckCapsule*和*VerifyReEnc*的验证计算工作放在区块链上完成是本文方案的优越性体现.

Table 1 Comparison of Privacy-Preserving Healthcare Systems

表 1 具有隐私保护的医疗系统对比

方案	隐私性	病患为中心	最小暴露原则	可验证性	可审计性	自治性
Thilakanathan 等人[14]	√	×	×	×	×	×
Zhou 等人[16]	√	√	×	×	×	√
Zhang 等人[10]	√	√	√	×	×	×
Sharma 等人[17]	√	√	×	√	×	×
本文方案	√	√	√	√	√	√

Table 2 Comparison with Related PRE Schemes

表 2 与相关的代理重加密方案对比

方案	*Enc*	*CheckCapsule*	*ReKeyGen*	*ReEnc*	*VerifyReEnc*	*Dec*
Wang 等人[27]	$5T_E$		$5T_E$	T_E+T_P		$4T_P$
Agyekum 等人[19]	$2T_E+T_P$		$2T_E+T_P$	T_P		T_P
Sun 等人[28]	$6T_E+T_P$	$6T_P$	$7T_E+T_P$	T_E+2T_P		$6T_E+8T_P$
Zhou 等人[16]	$6T_E+3T_P$		$5T_E+2T_P$	T_E+T_P		$2T_P$
UMBRAL[29]	$3T_E$	$2T_E$	$(2n+3)T_E$	$4T_E$	$(4+2n)T_E$	$(2t+3)T_E$
本文方案	$3T_E$	$2T_E$	$2T_E$	$4T_E$	$4T_E$	$2T_E$

注：T_E是椭圆曲线指数运算的开销，T_P是双线性配对运算的开销，n 是分布式代理的个数，t 是对应的门限值.

7 结 语

面对新时代智慧医疗中电子票据的数据共享及隐私保护问题，本文提出一种基于区块链和代理重加密的分布式数字身份(DID)方案. 区块链提供医疗数据的共享平台，其智能合约实现对密态数据进行自动化验证和审核. 借助代理重加密体制实现可验证凭证的安全流转，实现以病患为中心医疗模式. 此外，使用 BLS 聚合签名技术实现 DID 可验证表达，确保医疗数据的最小暴露原则. 最后，本文就方案给出详细的安全证明和实验说明，并就相关论文和方案做对比，充分体现本文方案的高效性及可行性. 在未来，我们将本文的框架应用到其他联盟链中，以满足医疗场景对于许可准入和高交易吞吐率的需求.

作者贡献声明：宋文鹏提出算法思路和应用场景并撰写论文；张亮参与设计实验并修改论文；马宇航进行相关实验并参与方案修改.

参 考 文 献

[1] Saxena A, Sharma T, Gupta H, et al. Revamping healthcare with Web 3.0[C]//Proc of the Int Conf on Innovative Computing & Communication (ICICC). Berlin: Springer, 2022. http://dx.doi.org/10.2139/ssrn.4381964

[2] Garzon S R, Yildiz H, Küpper A. Decentralized identifiers and self-sovereign identity in 6G[J]. IEEE Network, 2022, 36(4): 142−148

[3] Avellaneda O, Bachmann A, Barbir A, et al. Decentralized identity: Where did it come from and where is it going?[J]. IEEE Communications Standards Magazine, 2019, 3(4): 10−13

[4] Reed D, Sporny M, Longley D, et al. Decentralized identifiers (DIDs) v1.0. Draft Community Group Report[S/OL]. 2020 [2024-03-24]. https://www.w3.org/TR/did-core/#references

[5] Szalachowski P. Password-authenticated decentralized identities[J]. IEEE Transactions on Information Forensics and Security, 2021, 16: 4801−4810

[6] Costan V, Devadas S. Intel SGX explained[J/OL]. Cryptology ePrint Archive. 2016 [2024-03-24]. https://eprint.iacr.org/2016/086

[7] Fathalla E S, Azab M, Xin Chunsheng, et al. PT-SSIM: A proactive, trustworthy self-sovereign identity management system[J]. IEEE Internet of Things Journal, 2023, 10(19): 17155−17169

[8] Maram S K D, Zhang Fan, Wang Lun, et al. CHURP: Dynamic-committee proactive secret sharing[C]//Proc of the 2019 ACM SIGSAC Conf on Computer and Communications Security. New York ACM, 2019: 2369−2386

[9] Khalili M, Dakhilalian M, Susilo W. Efficient chameleon Hash functions in the enhanced collision resistant model[J]. Information Sciences, 2020, 510: 155−164

[10] Zhang Liang, Kan Haibin, Huang Honglan. Patient-centered cross-enterprise document sharing and dynamic consent framework using consortium blockchain and ciphertext-policy attribute-based encryption[C]//Proc of the 19th ACM Int Conf on Computing Frontiers. New York: ACM, 2022: 58−66

[11] Rouselakis Y, Waters B. Efficient statically-secure large-universe multi-authority attribute-based encryption[C]//Proc of Int Conf on Financial Cryptography and Data Security. Berlin: Springer, 2015: 315−332

[12] Wood G. A secure decentralised generalised transaction ledger[J]. Ethereum Project Yellow Paper, 2014, 151: 1−32

[13] Benet J. IPFS-content addressed, versioned, P2P file system[J]. arXiv preprint, arXiv: 1407.3561, 2014

[14] Thilakanathan D, Chen S, Nepal S, et al. A platform for secure monitoring and sharing of generic health data in the cloud[J]. Future Generation Computer Systems, 2014, 35: 102−113

[15] Ge C, Susilo W, Baek J, et al. A verifiable and fair attribute-based proxy re-encryption scheme for data sharing in clouds[J]. IEEE Transactions on Dependable and Secure Computing, 2021, 19(5): 2907−2919

[16] Zhou Yuyang, Zhao Liang, Jin Yuqiao, et al. Backdoor-resistant identity-based proxy re-encryption for cloud-assisted wireless body area networks[J]. Information Sciences, 2022, 604: 80−96

[17] Sharma B, Halder R, Singh J. Blockchain-based interoperable healthcare using zero-knowledge proofs and proxyre-encryption[C]// Proc of 2020 Int Conf on Communication Systems & Networks (COMSNETS). Piscataway, NJ: IEEE, 2020: 1−6

[18] Gao Juntao, Yu Haiyong, Zhu Xiuqin, et al. Blockchain-based digital rights management scheme via multiauthority ciphertext-policy attribute-based encryption and proxy re-encryption[J]. IEEE Systems Journal, 2021, 15(4): 5233−5244

[19] Agyekum K O B O, Xia Qi, Sifah E B, et al. A proxy re-encryption approach to secure data sharing in the Internet of things based on blockchain[J]. IEEE Systems Journal, 2021, 16(1): 1685−1696

[20] Yao Shimao, Dayot R V J, Ra I H, et al. An identity-based proxy re-encryption scheme with single-hop conditional delegation and multi-hop ciphertext evolution for secure cloud data sharing[J]. IEEE Transactions on Information Forensics and Security, 2023, 18: 3833−3848

[21] Boldyreva A. Threshold signatures, multisignatures and blind signatures based on the gap-Diffie-Hellman-group signature scheme[C]//Proc of the Int Workshop on Public Key Cryptography. Berlin: Springer, 2002: 31−46

[22] Boneh D, Lynn B, Shacham H. Short signatures from the Weil pairing[C]//Proc of the Int Conf on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2001: 514−532

[23] Keshta I, Aoudni Y, Sandhu M, et al. Blockchain aware proxy re-encryption algorithm-based data sharing scheme[J]. Physical Communication, 2023, 58: 102048

[24] Diffie W, Hellman M E. New directions in cryptography[J]. IEEE Transactions on Information Theory, 1976, 22(6): 365−390

[25] Rahimi A, Maddah-Ali M A. Multi-Party proof generation in QAP-based zk-SNARKs[J]. IEEE Journal on Selected Areas in Information Theory, 2021, 2(3): 931−941

[26] Karati A, Islam S K H, Biswas G P. A pairing-free and provably secure certificateless signature scheme[J]. Information Sciences, 2018, 450: 378−391

[27] Wang Xuan, Ma Jianfeng, Xhafa F, et al. Cost-effective secure E-health cloud system using identity based cryptographic techniques[J]. Future Generation Computer Systems, 2017, 67: 242−254

[28] Sun Maosheng, Ge Chunpeng, Fang Liming, et al. A proxy broadcast re-encryption for cloud data sharing[J]. Multimedia Tools and Applications, 2018, 77: 10455−10469

[29] Nunez D. UMBRAL: A threshold proxy re-encryption scheme[J/OL]. [2024-05-16]. https://raw.githubusercontent.com/nucypher/umbral-doc/master/umbral-doc.pdf



Song Wenpeng, born in 1982. PhD candidate. His main research interests include proxy re-encryption and blockchain.

宋文鹏, 1982 年生. 博士研究生. 主要研究方向为代理重加密、区块链.



Zhang Liang, born in 1989. PhD, associate professor, PhD supervisor. His main research interests include applied cryptography and blockchain.

张　亮, 1989 年生. 博士, 副教授、博士生导师. 主要研究方向为应用密码学、区块链.



Ma Yuhang, born in 2000. Master candidate. His main research interest includes cross-chain technology for blockchain.

马宇航, 2000 年生. 硕士研究生. 主要研究方向为区块链跨链技术.