计算机科学
COMPUTER SCIENCE

云雾泄露抵抗的智慧医疗安全认证协议

杨歆, 郭奕旻

引用本文

杨歆, 郭奕旻. 云雾泄露抵抗的智慧医疗安全认证协议[J]. 计算机科学, 2026, 53(4): 454-468.

YANG Xin, GUO Yimin. Smart Medical Secure Authentication Protocol for Cloud and Fog Leakage Resistance [J]. Computer Science, 2026, 53(4): 454-468.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.250100087

云雾泄露抵抗的智慧医疗安全认证协议

杨　歆　　郭奕旻
中南财经政法大学信息工程学院　武汉 430073
(xinyang@stu.zuel.edu.cn)

摘　要　智慧医疗在提升人们生活便捷性的同时,也带来了海量医疗数据在开放无线网络通信环境中的安全传输难题,这些数据在传输过程中易受多种内外部攻击的威胁。为确保医疗数据能够及时且有效地传输,云雾架构作为智慧医疗领域广泛采用的网络通信架构,通过雾计算对云计算的有效扩展,大幅缩短了云与终端设备之间的通信距离,有效缓解了因距离过大而产生的网络延迟与抖动问题。然而,现有基于云雾架构的通信方案大多采用单云-多雾-多设备集中式架构,这种设计易引发单点失效的风险。更严重的是,这些方案往往默认云是完全可信的,而现实中,云服务器同样面临内部攻击的风险,使得攻击者能在身份认证密钥协商阶段计算出会话钥,进而导致通信数据隐私泄露,严重影响通信安全。针对上述通信安全挑战,提出了一个抗云雾泄露攻击的智慧医疗安全认证密钥协商协议,利用区块链技术保障协议数据的安全性,在抵抗各种已知攻击的同时,还能够抵抗云雾泄露攻击。使用扩展的随机预言机模型(Random Oracle Model)证明了提出协议的语义安全性,使用启发式安全分析方法展示了所提协议实现了所有8个安全属性,同时,基于AVISPA安全分析工具验证了提出的协议是安全的。性能分析表明,相较于现有相关协议,提出的协议通信量较小,计算代价更小,能源消耗更低,且能抵抗更多的安全攻击。

关键词:智慧医疗;云雾泄露攻击;雾计算;认证;区块链
中图分类号　TP393

Smart Medical Secure Authentication Protocol for Cloud and Fog Leakage Resistance

YANG Xin and GUO Yimin
School of Information Engineering,Zhongnan University of Economics and Law,Wuhan 430073,China

Abstract　While smart healthcare enhances the convenience of people's lives,it also poses significant challenges for the secure transmission of massive medical data in open wireless network communication environments. These data are susceptible to various internal and external attacks during transmission. To ensure timely and effective medical data transmission,the cloud-fog architecture,widely adopted in smart healthcare for network communication,significantly shortens the communication distance between the cloud and terminal devices through the effective extension of cloud computing by fog computing,thereby effectively mitigating network latency and jitter issues caused by excessive distance. However,most existing authentication and communication schemes based on the cloud-fog architecture adopt a centralized architecture of single-cloud,multiple-fogs and multiple-devices,which is prone to the risk of single-point failure. More seriously,these schemes often assume that the cloud is completely trustworthy,whereas in reality,cloud servers also face the risk of internal attacks,enabling attackers to compute session keys during the identity authentication and key agreement phase,leading to the leakage of communication data privacy and severely impacting communication security. In response to these communication security challenges,this paper proposes a secure authentication and key agreement protocol for smart healthcare that is resistant to cloud-fog compromise attacks. Leveraging blockchain technology to ensure the security of protocol data,this protocol can withstand various known attacks while also resisting cloud-fog leakage attacks. The semantic security of the proposed protocol is demonstrated using the extended Random Oracle Model. A heuristic security analysis method is employed to show that the proposed protocol satisfies all eight security properties. Additionally,the security of the proposed protocol is verified using the AVISPA security analysis tool. Performance analysis indicates that,compared

到稿日期:2025-01-14　返修日期:2025-03-11
基金项目:国家自然科学基金(62102453);湖北省自然科学基金(2025AFC108);教育部人文社会科学研究项目(22YJCZH217);中南财经政法大学中央高校基本科研业务费专项资金(202451416)
This work was supported by the National Natural Science Foundation of China(62102453),Hubei Provincial Natural Science Foundation(2025AFC108),Project of Humanities and Social Sciences Research Project of Chinese Ministry of Education(22YJCZH217) and Fundamental Research Funds for the Central Universities of Zhongnan University of Economics and Law(202451416).
通信作者:郭奕旻(yiminguo@zuel.edu.cn)

with existing related protocols，the proposed protocol has lower communication overhead，lower computational cost，lower energy consumption，and stronger resistance to security attacks.

Keywords　Smart healthcare，Cloud and fog compromise attack，Fog computing，Authentication，Blockchain

1　引言

物联网概念的提出至今已有二十多年的时间，其技术也逐渐被应用到人们的日常生活中，其中一项重要应用就是智慧医疗[1]。伴随着物联网应用所产生的海量数据，云计算应运而生，它将大量的数据存储和计算任务分配给多个云服务器(Cloud Server，CS)处理，使得资源受限的本地设备也能够处理和分析大量数据[2]。然而，智慧医疗通信对于网络延迟十分敏感，它要求物联网医疗设备能够持续地、实时地、安全地监测患者的各项生理指标，以便医护人员远程访问患者数据，实现及时有效的干预、诊断和治疗[3]。同时，医疗领域对于信息泄露具有高度的敏感性，这些信息涉及个人隐私、健康状况乃至生命安全，一旦泄露将给患者带来难以估量的后果，因此数据的安全性在智慧医疗中尤为重要。而云计算通常通过远程网络来访问服务器，因此会存在网络延迟、网络中断等问题，进而影响医疗服务的效率[4]。雾计算的出现扩展了云计算的概念[5]，它具有低延迟、低能耗、高安全、高效率、位置感知、可扩展性、支持移动性等特点[6-7]。值得注意的是，雾层位于云层与物联网终端设备之间，它并非替代了云层[8]，而是将云的一些基本服务，如计算、网络、存储等服务，扩展到了网络边缘的雾节点。雾层中通常包含许多雾节点，而雾节点充当了一种靠近终端设备且能够在一定物理范围内快速响应与维护终端数据隐私，同时与云端进行通信的服务器。云雾架构是当前智慧医疗领域中广泛采用的一种网络通信架构，旨在确保医疗数据能够迅速且高效传输。

智慧医疗传输的数据蕴含高度敏感性，一旦泄露将给用户带来难以估量的后果。尽管智慧医疗显著提升了生活便捷性，但在开放的无线网络通信环境中，面对不断演化的攻击手段，如何确保海量医疗数据的隐私性与安全性已成为亟待解决的核心议题。为保障智慧医疗数据通信的安全与隐私，当前研究普遍聚焦于为智慧医疗传输架构中的参与方设计安全身份认证与密钥协商协议。

1.1　问题和动机

目前已提出许多基于云雾架构的身份认证密钥协商协议，这些协议大多遵循单云-多雾-多设备的集中式网络模型设计。鉴于雾层独特的计算环境，雾节点通常被视为不完全可信[9]，其存储信息易泄露。为应对雾节点泄露攻击，这些协议往往默认云服务器完全可信，并将用户及雾节点的相关秘密信息存储在云服务器，以确保认证密钥协商协议过程中这些秘密不会被攻击者直接获取。然而，云服务器也可能由第三方部署，存在被攻击的情况。一旦其中存储的秘密信息被泄露，不仅会严重威胁参与实体的隐私性，攻击者还可能破解协商产生的会话密钥，影响通信安全。因此，设计一个在云雾节点泄露时仍能确保安全的认证密钥协商协议，成为当前亟待解决的关键问题与挑战，具体归纳如下。

1)云雾泄露。现有基于云雾计算的认证协议通常默认云服务器为完全可信实体，而雾节点被视为完全可信或半可信的[10-11]。尽管部分协议能够抵御针对雾节点的攻击[12]，但并未充分考虑到云节点可能遭受的泄露攻击。

2)单点失效。现有的许多架构都属于单云-多雾-多设备集中式架构，即由唯一的云服务提供商负责管理雾节点与终端设备，容易出现单点失效、通信开销大、延迟高等问题，进而影响整个通信的运行[13-14]。

1.2　挑战和贡献

基于智慧医疗应用场景，设计在不完全可信云雾环境下高效、安全的认证协议是一项挑战性任务。它需要确保以下特性。

1)基本安全特性：认证协议应当具备匿名性和不可追踪性，以及密钥前向/后向安全性，能够抵抗去同步攻击、重放攻击、中间人攻击、假冒攻击、物联网设备捕获攻击、特权内幕攻击，以及短暂秘密泄露攻击。

2)关键安全属性：认证协议可以抵抗云、雾泄露攻击，避免单点失效问题。

近年来，区块链技术的出现与发展带来了新的设计思路。面对实时性、安全性要求较高以及终端设备资源受限的智慧医疗应用场景，区块链凭借其去中心化、分布式、数据共享、防篡改、可追溯等特点而拥有独特的优势[15-16]，设计基于区块链的云雾架构安全认证协议更能满足当下智慧医疗通信的安全需求。

为了抵御云雾泄露攻击带来的安全问题以及满足智慧医疗的实际需求，本文采用区块链技术，提出了一个分布式架构的智慧医疗安全认证协议。所有实体的注册信息会被上传到区块链，多个安全注册的云服务器和雾节点作为区块链网络中的节点，共同管理、维护着一个存储协议中实体信息的账本，其中云服务器可以访问和链入区块，雾节点只有访问区块的权限。即使某个雾节点出现故障，其余节点仍然可以继续处理网络数据。同时，在云雾数据被泄露的情况下，协议也能保障会话密钥的语义安全性。本文的主要贡献如下：

1)提出了一个抗云雾泄露的智慧医疗安全认证协议，协议采用分布式架构，实现了物联网医疗设备、雾节点和云服务器之间的相互认证与密钥协商。即使在云服务器存储信息完全泄露的情况下，仍能保障会话密钥的安全性，有效抵抗各种已知安全攻击，特别是云雾泄露攻击。

2)协议主要采用椭圆曲线和二元对称多项式密码原语，仅在物联网医疗设备和雾节点端使用椭圆曲线点乘运算，旨在最大限度地降低协议的计算开销，以适应智慧医疗的实时性需求。

3)扩展了 ROR(Real-or-Random)模型，以精确刻画攻击者对云服务器实施的泄露攻击，并在该模型下证明了所提出协议的安全性。通过启发式安全分析，进一步表明所提出的协议能够满足全部 8 种安全属性。此外，还采用了互联网安全协议和应用程序的自动验证工具(Automated Validation of

Internet Security Protocols and Applications, AVISPA)[17],对所提出协议的安全性进行了验证。与其他现有协议的详细性能分析和对比结果表明,所提出的协议在实现更多功能和安全特性的同时,还具有较低的通信开销、计算成本和能源消耗,即使在未知攻击场景下,其通信开销也显著低于其他协议。

2 相关工作

近年来,在智慧医疗和与其类似的应用场景下已经出现了许多认证协议,旨在为需要相互通信的实体之间建立一个共同的会话密钥,用于后续通信。根据协议采用的密码学工具,已有的认证协议大体可以分为三大类:使用哈希、异或和对称加密等方法的轻量级认证协议,使用公钥密码学方的认证协议,以及引入区块链的分布式架构认证协议。

早在 2016 年,Ibrahim[18]就提出了为边缘雾用户相互认证的协议,该协议只在注册时使用公钥加密方法,在认证阶段只执行了较少的哈希调用和对称加密/解密操作。然而,该协议在雾服务器中为每个雾用户存储一个密钥,因此容易受到雾服务器泄露攻击,且其网络模型采用单云-多雾的集中式架构,存在云的单点失效问题。Srinivas 等[19]提出了一个云环境下的可穿戴式医疗保健监测的身份认证协议,在相互认证时,只使用安全的单向哈希函数、模余操作和中国剩余定理等密码学原语,但他们也采用了以云为中心的结构。Wazid 等[20]为雾计算环境设计了一种只使用单向加密哈希函数和位异或的安全密钥管理及用户认证协议 SAKA-FC。Guo 等[21]的轻量级协议采用单云-多设备架构,协议面临着匿名性、前向保密性缺失的问题。他们在后续研究[22]中解决了这些问题,通过在验证器中嵌入动态身份,并添加一些额外的交换消息,实现了匿名、同步和完美前向安全性的目标。Guo 等[6]还提出了一种雾环境下基于二元对称多项式概念的智能家居远程认证协议,该协议提高了认证效率,同时还可以避免网关被入侵导致的各种攻击。

尽管采用哈希和对称加密算法的轻量级认证协议具有较小的计算量,但通常难以抵抗会话密钥泄露、离线字典猜测等攻击,部分学者基于公钥加密,提出了安全性能更高的认证协议。Jia 等[23]使用双线性配对,为雾驱动的医疗保健通信设计了一个认证密钥协商协议,并在 ROM 模型中证明了他们的协议是安全的,可以抵御众所周知的安全攻击。但是,该协议对雾节点假冒攻击没有抵御能力,并且不提供用户和雾节点匿名,单云的参与也使得身份认证效率变低,且存在单点故障的隐患。Ma 等[24]为了构建能够部署在实时和高速移动的应用环境中的认证协议,基于 Jia 等[23]的工作,为车联网设计了一种单云-多雾集中式架构的无双线性配对协议。然而,该协议中雾节点和车辆都不是完全可信的参与者,且注册权威给车辆用户传递的智能卡中还存储了其身份信息,使得其存在设备捕获攻击、智能卡被盗攻击和雾节点泄露攻击的问题。Li 等[25]也基于 Jia 等[23]的方案,利用椭圆曲线密码学(Elliptic Curve Cryptography, ECC)技术代替双线性配对操作,此外还使用了哈希操作。但该协议同样采用由一个可信云(Trusted Cloud, TC)参与系统初始化、注册和认证过程的集中式架构。Shen 等[26]提出了一种基于矩阵的单云-多雾密钥协议来支持密钥计算中的多方通信。然而,他们在协议中同样使用了计算成本昂贵的双线性配对操作。Kalaria 等[27]的协议使用哈希和 ECC,但也由一个 TC 作为注册权威参与系统初始化、注册和认证过程,因此容易出现传输延迟、单点失效等问题,也无法抵抗特权内幕攻击。Yao 等[28]使用哈希和 ECC,设计了一个车联网匿名认证方案。Ma 等[29]在研究中发现,Wang 等[30]提出的协议容易受到短暂秘密泄露(Ephemeral Secret Leakage, ESL)攻击,因此,他们使用哈希和 ECC,为无线局域网设计了一个更安全的身份认证协议。

可以看出,大多数轻量级认证协议和公钥认证协议都采用单云集中式架构,这容易导致传输延迟和单点失效问题。区块链是近年研究和应用十分广泛的新兴技术,具有去中心化、防篡改、可追溯等优势,其去中心化的特点符合物联网环境的分布式特征,能够有效解决单点失效问题。因此,近年来已有部分基于区块链技术的分布式架构认证方案被提出。Xu 等[13]的协议使用区块链技术,由多个可信权威(Trusted Authority, TA)共同管理和存储车辆相关信息的分类账。然而,该协议不支持匿名性。Xie 等[31]也为车联网提供了一个基于区块链的认证协议,该协议中车辆在认证阶段只需要执行轻量级的加密操作,例如哈希和对称加密,既实现了匿名跨域互切换认证,还能抵抗各种已知的攻击。Yu 等[32]使用区块链技术,提出了一种为自组织机载网络中运行无人机(Unmanned Aerial Vehicles, UAVs)设计的轻量级安全认证协议。Dong 等[33]提出了一种基于区块链的工业互联网跨域认证协议。Wei 等[34]、Shao 等[35]分别提出了基于区块链的跨域认证协议和远程医疗认证协议。

然而现有研究中的认证协议,无论是基于轻量级密码原语、公钥加密算法,还是区块链分布式架构的方案,在有云服务器参与的情况下,基本默认云服务器是完全可信的。但在现实中,云服务器可能由第三方运营,存在不完全可信的问题。一旦云服务器数据发生泄露,实体间的认证过程将面临严重的安全隐患。

3 预备知识

3.1 区块链

区块链由一系列的块连接而成,其中每个块由一个块头和块有效负载组成。此外,每个块头包含块版本、前一个或父块的哈希、当前块哈希、Merkle 树根、时间戳和块的所有者,每个块的有效负载还包含一个完整的事务列表[36]。区块链由于其分散化、不变性、透明度以及分布式点对点(Peer to Peer, P2P)网络中的持久性属性,而在医疗、金融等领域得到广泛应用。区块链有公共区块链、私有区块链和联盟区块链 3 种类型。

为了在区块链中添加块,需要一个共识算法。其中“共识”是指分布式 P2P 网络中不受信任的节点之间达到了准确一致的状态。对于区块链网络这种去中心化且节点不完全可信的环境,没有一个中心节点来确保其余节点的安全性与可信度,要达成共识是一件非常困难的任务。因此,共识算法作为一种确保不同节点账本一致的决策性算法,是区块链不可

或缺的一部分。区块链中，比较常用的两种共识机制，分别是工作量证明（Proof-of-Work，PoW）和权益证明（Proof-of-Stake，PoS）算法。

3.2　椭圆曲线密码学

3.2.1　椭圆曲线及其性质

有限域（伽罗华域）$GF(q)$上的椭圆曲线$y^2=x^3+ux+v \pmod q$是所有满足$y^2\equiv x^3+ux+v \pmod q$的解$(x,y)$的集合$E_q(u,v)$，其中，$q$是一个大素数，且$u,v\in Z_q=\{0,1,\cdots,q-1\}$，它具有一个无穷点或零点$O$。如果满足$4u^3+27v^2\neq 0 \pmod q$，就称这个椭圆曲线是非奇异的。椭圆曲线的一些性质如下。

1）如果$E_q(u,v)$上的点$A(x_1,y_1)$和$B(x_2,y_2)$满足$A+B=O$，那么$x_2=x_1$，$y_2=-y_1$。A和B互称为对方的加性逆。

2）对$\forall A\in E_q(u,v)$，有$A+O=O+A=A$。$E_q(u,v)$与单位元素O在模q加法运算下形成一个可交换的阿贝尔群。

3）$E_q(u,v)$上点的数量N满足不等式$q+2-2\sqrt{q}\leqslant N\leqslant q+1+2\sqrt{q}$。

4）椭圆曲线点加法：对于$E_q(u,v)$上的两点$A(x_1,y_1)$和$B(x_2,y_2)$，$R(x_3,y_3)=A+B$计算为$x_3=(k^2-x_1-x_2) \pmod q$，$y_3=(k(x_1-x_3)-y_1) \pmod q$。当$A=B$时，$k=(3x_1^2+u)/(2y_1) \pmod q$；当$A\neq B$时，$k=(y_2-y_1)/(x_2-x_1) \pmod q$。值得注意的是，当$A=B$时，$A+B=2\cdot A$可以称为点加倍操作。

5）椭圆曲线点乘法：ECC标量乘法表示为$s\cdot P$，其中s是一个标量值，P是椭圆曲线上的一个点。点乘法使用重复的点加倍操作和点加运算实现。

3.2.2　椭圆曲线计算难题

椭圆曲线在数学上有3种众所周知的计算难题，分别是椭圆曲线离散对数问题（Elliptic Curve Discrete Logarithm Problem，ECDLP）、椭圆曲线计算迪菲-赫尔曼问题（Elliptic Curve Computational Diffie-Hellman Problem，ECCDHP）以及椭圆曲线决策迪菲-赫尔曼问题（Elliptic Curve Decisional Diffie-Hellman Problem，ECDDHP）。

下面介绍本文设计协议时主要依据的ECDLP和ECCDHP。

定义1（ECDLP）　设P是$E_q(u,v)$上的一个点，且$Q=s\cdot P$，其中$s\in Z_q$。若已知P和Q的值，则存在一个大于且无限趋近于零的数ϵ，使得在概率多项式时间内计算出s的优势$Adv\leqslant\epsilon$。也就是说，在概率多项式时间内计算s是十分困难的。

定义2（ECCDHP）　设P是$E_q(u,v)$上的一个点，并且给定两点$x\cdot P$和$y\cdot P$，其中$x,y\in Z_q$，则在概率多项式时间内计算出$(x\cdot y)\cdot P$的值是困难的。

3.3　物理不可克隆函数

物理不可克隆函数（Physical Unclonable Function，PUF）是一种硬件安全技术，它使用固有设备结构来生成对给定输入的无法克隆的唯一设备响应。PUF的输入和输出对被称为挑战-响应对，表示为$R=PUF(C)$，其中，C表示挑战，R表示响应。PUF具有以下特点。

1）物理不可克隆性：PUF的输出值取决于设备的物理微观结构，也就是说，对于安装了PUF的不同设备，相同的输入会产生不同的结果。

2）可重复性和唯一性：对于一个PUF，相同的输入产生相同的输出，而不同的输入产生不同的输出。

3）不可预测性：对于任何PUF，在多项式时间内预测给定挑战的响应都是不可行的。

3.4　t度二元对称多项式

有限域$GF(q)$上定义的t度二元对称多项式表示为$f(x,y)=\sum_{i=0}^{t}\sum_{j=0}^{t}a_{ij}x^iy^j$，其中，$x$和$y$为两个变量，系数$a_{ij}\in Z_q$，并且多项式满足对称性，即$f(x,y)=f(y,x)$。

如果给定一个元素$m\in Z_q$，则称$f(m,y)$是变量y的一个t度单变量多项式。当$y=s$时，计算多项式的值$f(m,s)$需要进行t次模加法和t次模乘法。

4　系统模型

4.1　网络模型

本文的网络模型如图1所示，网络分为3层，分别是医疗设备层、雾层以及云层。网络模型中所涉及实体的各种角色如下。

医疗设备（Medical Devices，MDs）：智慧医疗中的智能医疗设备，可以在一个特定的应用程序中安装或部署多个医疗设备。医疗设备通常负责收集患者及其周围的信息，并将其传输到附近的雾节点进行进一步处理。医疗设备通常在内存、存储、通信和计算能力方面资源有限。

雾节点（Fog Nodes，FNs）：FNs被认为是不完全可信的，它们负责与其覆盖范围内的医疗设备进行相互认证。此外，FN将从其对应的医疗设备MDs中收集数据，并形成包含交易的块。然后，将所构建的块转发到其关联的云服务器。

云服务器（Cloud Servers，CSs）：一组云服务器形成了一个点对点的CS网络。它们主要负责使用区块链共识算法验证其关联的雾节点转发的块，并将区块添加到区块链中。一旦一个块被添加到区块链中，就不允许修改或删除。

注册权威（Registration Authorities，RAs）：网络中完全受信任的一些实体，负责注册网络中所有已部署的医疗设备、雾节点和云服务器。RAs通过共识机制将注册实体的关键信息打包上链，而每个实体在部署或放置在物联网环境之前，都要预加载适当的凭据。

区块链网络（Blockchain Network，BN）：在本文协议中，区块链网络采用联盟区块链的方式，由所有网关节点使用PoS算法共同维护。其中，一组云服务器形成了一个点对点的CS网络，即区块链网络中的主体节点负责验证和添加区块到现有区块链中。雾节点负责收集医疗设备中的数据，形成包含交易的区块并提交到对应的云服务器中，并且只能读取区块链中的交易。区块链网络的这种分布式架构使得多个节点相互制衡，共同作用，能有效防止单点失效问题出现。

综上所述，本文网络模型中云层包含所有云服务器，每个云服务器都关联一些雾节点，这些雾节点构成雾层。每个雾节点又关联医疗物联网中的许多医疗设备，所有医疗设备一起组成医疗设备层。区块链网络则由云层、雾层以及RA共同参与，其中，RA只在注册过程访问区块链。

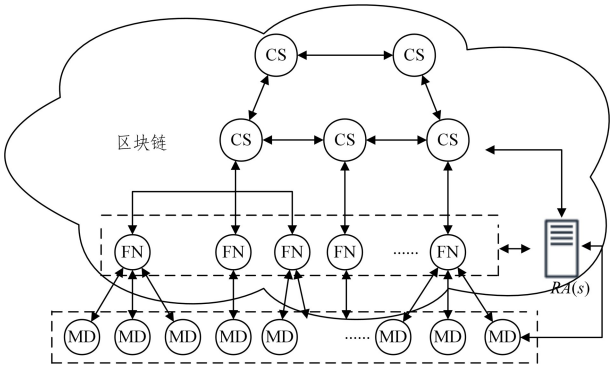


图 1 网络模型

Fig. 1 Network model

4.2 威胁模型

针对本文网络模型，所考虑的威胁模型如下。

1)根据被广泛使用的 Dolev-Yao(DY)威胁模型[37]，假设敌手$\mathcal{A}$不仅可以拦截通信实体(即医疗设备、雾节点、云服务器)在公共信道上的通信信息，还可以修改或删除消息内容，甚至插入恶意信息，但其无法影响安全信道中传输的信息。

2)根据被广泛采用的 Canetti and Krawczyk(CK)-敌手模型[38]，敌手$\mathcal{A}$不仅可以像 DY 模型那样拦截、修改、删除或插入消息，还可以破坏设备存储的密钥、密钥凭证以及会话状态等信息。

3)医疗设备不被视为完全可信的实体。如果敌手俘获医疗设备，就能够采取一些攻击手段，如功率分析攻击[39]，从捕获设备的存储器中提取私密数据，还可以进一步用于发起其他攻击，例如对其他未受影响的设备进行模拟攻击。

4)假设雾节点和云服务器是不完全可信的实体，两个或两个以上的云服务器不会互相勾结。

5)注册权威 RAs 被视为完全可信，无法被敌手破坏。

4.3 评价标准

参考现有文献以及本文所做工作，为智慧医疗场景的认证协议设计了新的评价标准，具体如表 1 所列。这些标准可以分为可用性属性(EC1－EC2)和安全属性(EC3－EC8)。参考了文献[7,40-43]等，首先定义一些基本安全属性(EC3－EC6)。此外，基于云雾架构中认证协议面临的云雾泄露问题，增加了 EC7 与 EC8 作为本文设计中需要实现的关键安全属性。

表 1 评价标准

Table 1 Evaluation criteria

评价标准	属性	描述
EC1	相互认证	认证协议中的参与者需要实现相互身份认证
EC2	密钥协商	需要在认证协议中的参与者之间协商会话密钥
EC3	匿名和不可追溯性	敌手$\mathcal{A}$不能通过其能力获取实体的身份和行为
EC4	密钥前向/后向安全	$\mathcal{A}$无法根据已泄露的会话密钥推断出其之前或之后的会话密钥
EC5(1－6)	抵抗已知攻击	协议可抵抗如去同步攻击(EC5-1)、重放攻击(EC5-2)、中间人攻击(EC5-3)、假冒攻击(EC5-4)、特权内幕攻击(EC5-5)、短暂秘密泄露攻击(EC5-6)等攻击
EC6	抗医疗设备捕获	$\mathcal{A}$无法通过医疗设备捕获攻击获取会话密钥
EC7	抗雾节点泄露	$\mathcal{A}$无法通过雾节点泄露攻击获取会话密钥
EC8	抗云服务器泄露	$\mathcal{A}$无法通过云服务器泄露攻击获取会话密钥

5 提出的协议

本章提出了一种基于区块链抗云雾泄露攻击的智慧医疗安全认证密钥协商协议。协议包含 3 个阶段，即初始化阶段、注册阶段、身份认证与密钥协商阶段。当智慧医疗系统启动时，由 RA 进行初始化阶段。所有的 MD，FN 和 CS 在部署之前，都必须通过注册阶段在 RA 中进行注册。初始化阶段和注册阶段都是在安全的环境中进行的。在身份认证与密钥协商阶段，MD，FN 与 CS 完成相互认证和密钥协商。方案中涉及的符号和描述如表 2 所列。

表 2 符号及描述

Table 2 Notations and descriptions

符号	描述
$E_q(u,v)$	形如$y^2=x^3+ux+v(\mathrm{mod}\ q)$且$4u^3+27v^2\neq 0(\mathrm{mod}\ q)$的椭圆曲线
P	$E_q(u,v)$中的一个基点，阶与 q 一样大
$\mathrm{RA}, MD_i, FN_j, CS_k$	注册权威，医疗设备，雾节点，云服务器
$ID_{\mathrm{RA}}, ID_{MD_i}, ID_{FN_j}, ID_{CS_k}$	注册权威，医疗设备，雾节点，云服务器的真实身份
s, P_{pub}	RA 的私钥和公钥
$PID_{\mathrm{RA}}, PID_{MD_i}, PID_{FN_j}, PID_{CS_k}$	注册权威，医疗设备，雾节点，云服务器的伪身份
TID_{MD_i}, TID_{FN_j}	医疗设备和雾节点的临时身份
k_{FN_j}, k_{CS_k}	雾节点，云服务器的私钥
Pub_{FN_j}, Pub_{CS_k}	雾节点，云服务器的公钥
$f(x,y)$	伽罗华域 $GF(q)$上的 t 度二元对称多项式
r_i	随机数
$T_i, \Delta T$	时间戳，最大传输延迟
SK_i	会话密钥
SKV_i	会话密钥验证器
$h(\cdot)$	抗碰撞的加密单向哈希函数
$\|, \oplus$	连接操作，按位异或操作

5.1　初始化阶段

在此阶段，由 RAs 通过 PoS 共识机制推选出一个 RA 进行智慧医疗系统的初始化。

步骤 1　RA 首先选择一个非奇异椭圆曲线 $E_q(u,v)$，它是由一个大素数 q（规模至少大于 512 bit）所构成的具有一个无穷点或零点 O 的有限域（伽罗华域）$GF(q)$ 上的椭圆曲线，其形式是 $y^2=x^3+ux+v \pmod q$。其中，$4u^3+27v^2\neq 0 \pmod q$，且 $u,v\in Z_q=\{0,1,\cdots,q-1\}$。由该椭圆曲线上点的加法所构成的交换群应该是一个阶数很大的阿贝尔群。随后，RA 在椭圆曲线 $E_q(u,v)$ 上选择一个阶与 q 一样大的基点 P。RA 选择唯一身份 ID_{RA}，并随机生成一个随机数 $s\in Z_q^*$ 作为系统私钥，同时计算对应的公钥 $P_{\mathrm{pub}}=s\cdot P$。

步骤 2　RA 选择一个单向加密哈希函数 $h:\{0,1\}^*\rightarrow\{0,1\}^l$，即它能将一个任意长度的输入字符串 $x\in\{0,1\}^*$ 映射为一个固定长度为 l 位的输出字符串 $h(x)\in\{0,1\}^l$。例如，$h(\cdot)$ 可以被视为产生 160 位哈希值的安全哈希算法（SHA-1），为了更具安全性，它还可以是 SHA-256 或 SHA-512。RA 计算自己的伪身份 $PID_{\mathrm{RA}}=h(ID_{\mathrm{RA}}\|s)$。

步骤 3　RA 将 s 和 ID_{RA} 添加到区块链中，并设置仅 RAs 可以访问这个区块，并在网络中公布参数 $\{E_q(u,v),P,h(\cdot),PID_{\mathrm{RA}},P_{\mathrm{pub}}\}$。

5.2　注册阶段

此阶段由受信任的 RA 在安全环境下注册所有已部署的医疗设备 $MD_i(i=1,2,\cdots,n_{\mathrm{MD}})$，雾节点 $FN_j(j=1,2,\cdots,n_{\mathrm{FN}})$，以及云服务器 $CS_k(k=1,2,\cdots,n_{\mathrm{CS}})$。详细的注册过程如下。

1）医疗设备注册。对于每个已部署的医疗设备 MD_i，RA 为其选择一个唯一的真实身份 ID_{MD_i} 和对应的临时身份 TID_{MD_i}，令 $TID_{MD_i}^{\mathrm{old}}=\mathrm{null}$，$TID_{MD_i}^{\mathrm{new}}=TID_{MD_i}$。计算伪随机身份 $PID_{MD_i}=h(ID_{MD_i}\|s)$，选择随机挑战 C_i，计算 $R_i=PUF(C_i)$。RA 为 MD_i 加载注册凭证 $\{TID_{MD_i}^{\mathrm{old}},TID_{MD_i}^{\mathrm{new}},PID_{MD_i},C_i\}$，并通过 PoS 算法将 $\{PID_{MD_i},C_i,R_i\}$ 打包成区块，设置索引为 PID_{MD_i}，添加到现有区块链中。

2）雾节点注册。RA 为每个雾节点 FN_j 选择唯一身份 ID_{FN_j} 和一个临时身份 TID_{FN_j}，令 $TID_{FN_j}^{\mathrm{old}}=\mathrm{null}$，$TID_{FN_j}^{\mathrm{new}}=TID_{FN_j}$。计算其伪随机身份 $PID_{FN_j}=h(ID_{FN_j}\|s)$，公私钥对 $\{k_{FN_j}\in Z_q^*,Pub_{FN_j}=k_{FN_j}\cdot P\}$。RA 在 $GF(q)$ 生成一个形如“$f(x,y)=\sum_{i=0}^{t}\sum_{j=0}^{t}a_{ij}x^iy^j$”的 t 度二元对称多项式（其中，系数 $a_{ij}\in Z_q$，多项式满足 $f(x,y)=f(y,x)$，且 t 的值应该不小于网络中部署的雾节点数目），并计算出 FN_j 的多项式份额为 $f(PID_{FN_j},y)=\sum_{m=0}^{t}\sum_{n=0}^{t}a_{mn}PID_{FN_j}^m y^n \pmod q$。RA 在 FN_j 中存储 $\{TID_{FN_j}^{\mathrm{old}},TID_{FN_j}^{\mathrm{new}},PID_{FN_j},f(PID_{FN_j},y),k_{FN_j}\}$，以及 $\{(TID_{MD_i},PID_{MD_i})\mid i=1,2,\cdots,n_{\mathrm{MD}}\}$，其中 TID_{MD_i} 为 $TID_{MD_i}^{\mathrm{new}}$ 的值。RAs 将 Pub_{FN_j} 在网络中公开，通过 PoS 算法将 $\{PID_{FN_j}\}$ 打包成区块，并添加到区块链中。

3）云服务器注册。RA 为每个云服务器 CS_k 选择唯一身份 ID_{CS_k}，计算其伪随机身份 $PID_{CS_k}=h(ID_{CS_k}\|s)$，公私钥对 $\{k_{CS_k}\in Z_q^*,Pub_{CS_k}=k_{CS_k}\cdot P\}$。RA 选择 t 度二元对称多项式 $f(x,y)=\sum_{i=0}^{t}\sum_{j=0}^{t}a_{ij}x^iy^j$，计算 $f(PID_{CS_k},y)=\sum_{i=0}^{t}\sum_{j=0}^{t}a_{ij}PID_{CS_k}^i y^j \pmod q$。随后，RA 将 $\{PID_{CS_k},f(PID_{CS_k},y),k_{CS_k},Pub_{CS_k}\}$ 和 $\{(TID_{FN_j}^{\mathrm{old}},TID_{FN_j}^{\mathrm{new}},PID_{FN_j})\mid j=1,2,\cdots,n_{\mathrm{FN}}\}$ 存储在 CS_k 中，其中 Pub_{CS_k} 被设置为公开。RAs 通过 PoS 算法将 $\{PID_{CS_k}\}$ 打包成区块，并添加到区块链中。

5.3　身份认证和密钥协商阶段

此阶段将分别实现医疗设备与雾节点，以及雾节点与云服务器之间的身份认证与密钥协商协议，协商得到的共享会话密钥将被用于后续通信。算法流程如图 2 所示。

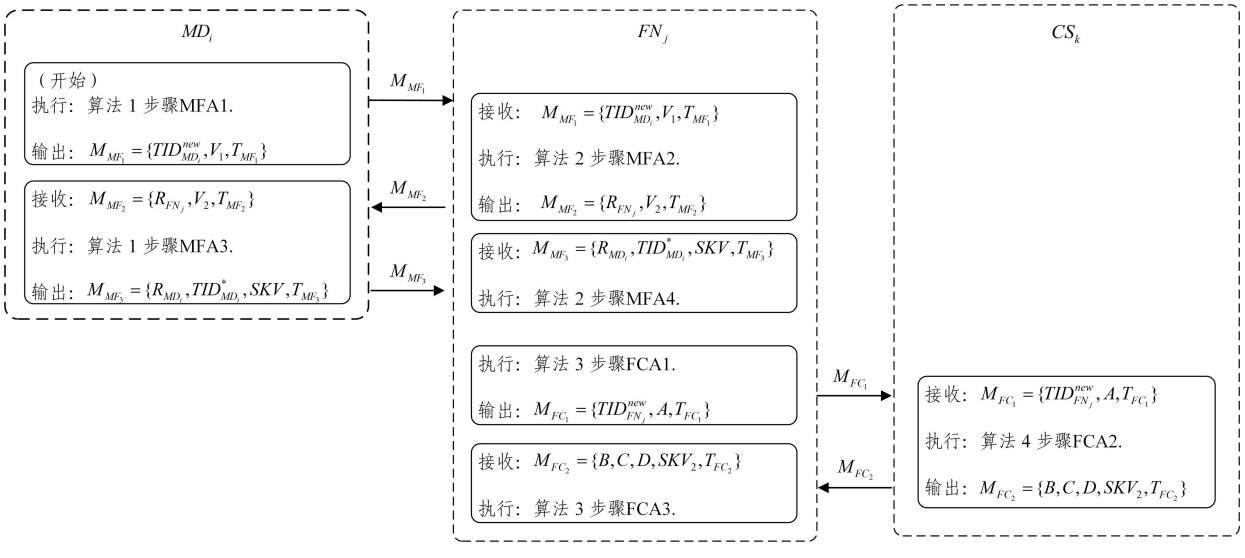


图 2　算法流程图

Fig. 2　Flowchart of the algorithm

5.3.1　MD_i 与 FN_j 之间的认证过程

在此阶段，一个已注册医疗设备 MD_i 将与其想要通信的 FN_j 进行相互认证与密钥协商，MD_i 和 FN_j 在这一阶段的算法流程分别如算法 1 和算法 2 所示。

步骤 MFA1　MD_i 随机生成时间戳 T_{MF_1}，根据存储的 C_i 计算 $R_i=PUF(C_i)$，并计算 $V_1=h(PID_{MD_i}\|C_i\|R_i\|T_{MF_1})$。随后，$MD_i$ 构造消息 $M_{MF_1}=\{TID_{MD_i}^{\mathrm{new}},V_1,T_{MF_1}\}$，并通过公共信道发送给 FN_j。

算法 1 医疗设备 MD_i 算法

输入：$\{R_{FN_j}, V_2, T_{MF_2}\}$

输出：$\{TID^{new}_{MD_i}, V_1, T_{MF_1}\}$，$\{R_{MD_i}, TID^{*}_{MD_i}, SKV, T_{MF_3}\}$ 或 false

1. 步骤 MFA1。
2. begin
3. 计算 $R_i = PUF(C_i)$，
4. 和 $V_1 = h(PID_{MD_i} \| C_i \| R_i \| T_{MF_1})$。
5. end
6. 步骤 MFA3。
7. /* 等待 FN_j 的消息 */
8. 从 FN_j 处收到 $M_{MF_2} = \{R_{FN_j}, V_2, T_{MF_2}\}$
9. begin
10. if($|T_2 - T_{MF_2}| < \Delta T$) then
11. if($V_2 \cdot P == R_{FN_j} + h(R_i \| T_{MF_2}) * Pub_{FN_j}$)
12. then
13. 生成 r_{MD_i} 和 T_{MF_3}，
14. 计算 $S_{MD_i} = h(r_{MD_i} \| PID_{MD_i} \| T_{MF_3})$，
15. $R_{MD_i} = S_{MD_i} \cdot P$
16. $SK_{ij} = h(S_{MD_i} \cdot R_{FN_j} \| R_i \| T_{MF_3})$
17. 计算 $TID'_{MD_i} = TID^{new}_{MD_i} \oplus h(SK_{ij})$
18. $TID^{old}_{MD_i} = TID^{new}_{MD_i}$，$TID^{new}_{MD_i} = TID'_{MD_i}$
19. $TID^{*}_{MD_i} = TID'_{MD_i} \oplus h(SK_{ij} \| T_{MF_3})$
20. $SKV = h(SK_{ij} \| TID'_{MD_i} \| T_{MF_3})$
21. end
22. end
23. else return false
24. end

步骤 MFA2 FN_j 在时间 T_1 收到 MD_i 的消息 M_{MF_1} 后，首先验证 $|T_1 - T_{MF_1}| < \Delta T$ 是否成立，其中 ΔT 是系统的“最大传输延迟”。若不成立，则直接终止当前会话；否则，FN_j 通过 $TID^{new}_{MD_i}$ 获取到对应的 PID_{MD_i}，然后访问区块链中索引为 PID_{MD_i} 的块，获取到 C_i 和 R_i 的值。验证 $h(PID_{MD_i} \| C_i \| R_i \| T_{MF_1})$ 是否和 V_1 的值相等，如果成立，FN_j 就成功验证了 MD_i，否则会话结束。随后，FN_j 生成一个随机的 $r_{FN_j} \in Z_q^*$ 和当前时间戳 T_{MF_2}，计算参数 $S_{FN_j} = h(r_{FN_j} \| PID_{FN_j} \| T_{MF_2})$，$R_{FN_j} = S_{FN_j} \cdot P$。之后，$FN_j$ 计算 $V_2 = S_{FN_j} + h(R_i \| T_{MF_2}) * k_{FN_j} \pmod q$，然后构造消息 $M_{MF_2} = \{R_{FN_j}, V_2, T_{MF_2}\}$，并通过公共信道发送到 MD_i。

算法 2 雾节点 FN_j 算法（与 MD_i 认证）

输入：$\{TID^{new}_{MD_i}, V_1, T_{MF_1}\}$，$\{R_{MD_i}, TID^{*}_{MD_i}, SKV, T_{MF_3}\}$

输出：$\{R_{FN_j}, V_2, T_{MF_2}\}$ 或 false

1. 步骤 MFA2。
2. /* 等待 MD_i 的消息 */
3. 从 MD_i 处收到 $M_{MF_1} = \{TID^{new}_{MD_i}, V_1, T_{MF_1}\}$
4. begin
5. if($|T_1 - T_{MF_1}| < \Delta T$) then
6. 通过 $TID^{new}_{MD_i}$ 获取对应的 PID_{MD_i}，访问区块链中索引为 PID_{MD_i} 的块以获取 C_i 和 R_i。
7. if($h(PID_{MD_i} \| C_i \| R_i \| T_{MF_1} = V_1$) then
8. 生成 r_{FN_j} 和 T_{MF_2}，
9. 计算 $S_{FN_j} = h(r_{FN_j} \| PID_{FN_j} \| T_{MF_2})$
10. $R_{FN_j} = S_{FN_j} \cdot P$，
11. $V_2 = S_{FN_j} + h(R_i \| T_{MF_2}) * k_{FN_j} \pmod q$.
12. end
13. end
14. else return false
15. end
16. 步骤 MFA4。
17. /* 等待 MD_i 的消息 */
18. 从 MD_i 处收到 $M_{MF_3} = \{R_{MD_i}, TID^{*}_{MD_i}, SKV, T_{MF_3}\}$
19. begin
20. if($|T_3 - T_{MF_3}| < \Delta T$) then
21. 计算 $SK_{ji} = h(S_{FN_j} \cdot R_{MD_i} \| R_i \| T_{MF_3})$，
22. $TID'_{MD_i} = TID^{*}_{MD_i} \oplus h(SK_{ij} \| T_{MF_3})$，
23. if($h(SK_{ji} \| TID'_{MD_i} \| T_{MF_3}) = SKV$) then
24. 更新 TID_{MD_i} 为 TID'_{MD_i}。
25. end
26. end
27. else return false
28. end

步骤 MFA3 在时间 T_2 收到 M_{MF_2} 后，MD_i 首先验证是否有 $|T_2 - T_{MF_2}| < \Delta T$。若不等式成立，则 MD_i 验证 $V_2 \cdot P = R_{FN_j} + h(R_i \| T_{MF_2}) * Pub_{FN_j}$ 是否成立。若验证成功，则 MD_i 也将 FN_j 视为合法节点。随后，MD_i 生成随机数 r_{MD_i} 和时间戳 T_{MF_3}，计算私有参数 $S_{MD_i} = h(r_{MD_i} \| PID_{MD_i} \| T_{MF_3})$、公有参数 $R_{MD_i} = S_{MD_i} \cdot P$ 和会话钥 $SK_{ij} = h(S_{MD_i} \cdot R_{FN_j} \| R_i \| T_{MF_3})$。之后计算新的临时身份 $TID'_{MD_i} = TID^{new}_{MD_i} \oplus h(SK_{ij})$，并更新 $TID^{old}_{MD_i} = TID^{new}_{MD_i}$，$TID^{new}_{MD_i} = TID'_{MD_i}$。然后，计算 $TID^{*}_{MD_i} = TID'_{MD_i} \oplus h(SK_{ij} \| T_{MF_3})$ 和会话密钥验证器 $SKV = h(SK_{ij} \| TID'_{MD_i} \| T_{MF_3})$，并通过开放信道发送消息 $M_{MF_3} = \{R_{MD_i}, TID^{*}_{MD_i}, SKV, T_{MF_3}\}$ 给 FN_j。

步骤 MFA4 FN_j 在时间 T_3 收到 M_{MF_3} 后，首先验证 $|T_3 - T_{MF_3}| < \Delta T$ 是否成立。若成立，则计算会话密钥 $SK_{ji} = h(S_{FN_j} \cdot R_{MD_i} \| R_i \| T_{MF_3})$ 和临时身份 $TID'_{MD_i} = TID^{*}_{MD_i} \oplus h(SK_{ij} \| T_{MF_3})$，并进一步验证 $h(SK_{ji} \| TID'_{MD_i} \| T_{MF_3}) = SKV$ 是否成立，如果成立，则表明密钥协商成功。FN_j 还需要将其数据库中 PID_{MD_i} 对应的 TID_{MD_i} 更新为 TID'_{MD_i}。

通过上述步骤，MD_i 与 FN_j 成功建立了共享的会话密钥 SK_{ij}（$= SK_{ji}$）。

5.3.2 FN_j 与 CS_k 之间的认证过程

在这一阶段，某个雾节点 FN_j 需要与其相关联的 CS_k 协商好一个共享会话密钥，以便后续将自己打包的块安全转发给 CS_k。FN_j 和 CS_k 的相关步骤流程如算法 3 和算法 4 所示。

算法 3 雾节点 FN_j 算法（与 CS_k 认证）

输入：$\{B, C, D, SKV_2, T_{FC_2}\}$

输出：$\{TID^{new}_{FN_j}, A, T_{FC_1}\}$ 或 false

1. 步骤 FCA1。
2. begin
3. 生成 r_{FN_j} 和 T_{FC_1}，
4. 计算 $h(r_{FN_j} \| k_{FN_j} \| PID_{FN_j} \| T_{FC_1})$，
5. $A = H_1 \oplus h(PID_{FN_j} \| T_{FC_1})$。
6. end

7. 步骤 FCA3。
8. /* 等待 CS_k 的消息 */
9. 从 CS_k 处收到 $M_{FC_2}=\{B,C,D,SKV_2,T_{FC_2}\}$
10. begin
11. if($|T_2-T_{FC_2}|<\Delta T$) then
12. 计算 $PID_{CS_k}=D\oplus h(TID_{FN_j}^{new}\|T_{FC_2})$，
13. $H_2=B\oplus h(f(PID_{FN_j},PID_{CS_k})\|T_{FC_2})$，
14. $SK_{jk}=h(H_1\|H_2\|f(PID_{FN_j},PID_{CS_k}))$，
15. $TID'_{FN_j}=C\oplus h(SK_{jk})$。
16. if($SKV_2==h(SK_{jk}\|TID'_{FN_j}\|T_{FC_2})$) then
17. $TID_{FN_j}^{old}=TID_{FN_j}^{new}$，$TID_{FN_j}^{new}=TID'_{FN_j}$。
18. end
19. end
20. else return false
21. end

步骤 FCA1　FN_j 随机生成 $r_{FN_j}\in Z_q^*$ 和时间戳 T_{FC_1}，计算 $H_1=h(r_{FN_j}\|k_{FN_j}\|PID_{FN_j}\|T_{FC_1})$ 和 $A=H_1\oplus h(PID_{FN_j}\|T_{FC_1})$。随后，$FN_j$ 通过公共信道发送消息 $M_{FC_1}=\{TID_{FN_j}^{new},A,T_{FC_1}\}$ 给 CS_k。

步骤 FCA2　在时间 T_1，CS_k 收到 M_{FC_1} 后，首先验证 $|T_1-T_{FC_1}|<\Delta T$ 是否成立。若验证通过，则接收 M_{FC_1}。随后，CS_k 通过得到的 $TID_{FN_j}^{new}$ 查找得到 PID_{FN_j}，在区块链中验证是否存在索引为 PID_{FN_j} 的块，若不存在则终止认证，然后通过 $A\oplus h(PID_{FN_j}\|T_{FC_1})$ 还原 H_1。CS_k 生成随机数 $r_{CS_k}\in Z_q^*$ 和当前时间戳 T_{FC_2}，计算 $H_2=h(r_{CS_k}\|k_{CS_k}\|T_{FC_2})$。然后，计算 $B=H_2\oplus h(f(PID_{CS_k},PID_{FN_j})\|T_{FC_2})$ 和与 FN_j 共享的密钥 $SK_{kj}=h(H_1\|H_2\|f(PID_{CS_k},PID_{FN_j}))$。计算完成后，$CS_k$ 为 FN_j 生成新的临时身份 TID'_{FN_j}，计算 $SKV_2=h(SK_{kj}\|TID'_{FN_j}\|T_{FC_2})$，$C=TID'_{FN_j}\oplus h(SK_{kj})$，以及 $D=PID_{CS_k}\oplus h(TID_{FN_j}^{new}\|T_{FC_2})$。先用 $TID_{FN_j}^{new}$ 更新自己数据库中对应于 PID_{FN_j} 的 $TID_{FN_j}^{old}$，再用 TID'_{FN_j} 更新 $TID_{FN_j}^{new}$。最后，CS_k 通过公共信道向 FN_j 发送信息 $M_{FC_2}=\{B,C,D,SKV_2,T_{FC_2}\}$。

步骤 FCA3　FN_j 在时间 T_2 收到 M_{FC_2} 后，首先验证是否有 $|T_2-T_{FC_2}|<\Delta T$。若不等式成立，则 FN_j 计算 $PID_{CS_k}=D\oplus h(TID_{FN_j}^{new}\|T_{FC_2})$，然后还原 $H_2=B\oplus h(f(PID_{FN_j},PID_{CS_k})\|T_{FC_2})$，计算共享密钥 $SK_{jk}=h(H_1\|H_2\|f(PID_{FN_j},PID_{CS_k}))$。然后，$FN_j$ 计算 $TID'_{FN_j}=C\oplus h(SK_{jk})$，并验证 $SKV_2=h(SK_{jk}\|TID'_{FN_j}\|T_{FC_2})$ 是否正确。若正确，FN_j 将 CS_k 视为合法节点，并将 SK_{jk} 作为后续通信的会话密钥。最后，FN_j 用 $TID_{FN_j}^{new}$ 更新 $TID_{FN_j}^{old}$，再用 TID'_{FN_j} 更新 $TID_{FN_j}^{new}$。

算法 4　云服务器 CS_k 算法

输入：$\{TID_{FN_j}^{new},A,T_{FC_1}\}$
输出：$\{B,C,D,SKV_2,T_{FC_2}\}$ 或 false
1. 步骤 FCA2。
2. /* 等待 FN_j 的消息 */
3. 从 FN_j 处收到 $M_{FC_1}=\{TID_{FN_j}^{new},A,T_{FC_1}\}$
4. begin
5. if($|T_2-T_{FC_2}|<\Delta T$) then
6. 通过 $TID_{FN_j}^{new}$ 获取对应的 PID_{FN_j}。
7. if(区块链中存在索引为 PID_{FN_j} 的块) then
8. 生成 r_{CS_k} 和 T_{FC_2}，
9. 计算 $H_2=h(r_{CS_k}\|k_{CS_k}\|T_{FC_2})$，
10. $B=H_2\oplus h(f(PID_{CS_k},PID_{FN_j})\|T_{FC_2})$，
11. $SK_{kj}=h(H_1\|H_2\|f(PID_{CS_k},PID_{FN_j}))$。
12. 生成 TID'_{FN_j}，
13. 计算 $SKV_2=h(SK_{kj}\|TID'_{FN_j}\|T_{FC_2})$，
14. $C=TID'_{FN_j}\oplus h(SK_{kj})$，
15. $D=PID_{CS_k}\oplus h(TID_{FN_j}^{new}\|T_{FC_2})$，
16. $TID_{FN_j}^{old}=TID_{FN_j}^{new}$，$TID_{FN_j}^{new}=TID'_{FN_j}$。
17. end
18. end
19. else return false
20. end

至此，FN_j 与 CS_k 成功建立了共享的会话密钥 SK_{jk}（$=SK_{kj}$）。

6　安全性证明

Wang 等[44]提出，在分析一个安全协议时，结合形式化安全证明与非形式化安全分析是必要的。对于形式化安全分析，目前广泛使用 ROR 预言机模型，但由于传统 ROR 模型并未包含云服务器泄露攻击的情况，因此本文扩展了 ROR 模型，并在扩展模型下进行形式化安全证明。同时，进行启发式安全分析，以证明提出的协议能够抵抗对被动和主动敌手的各种潜在攻击。最后，使用 AVISPA 工具进一步验证了本协议的安全性。

6.1　扩展 ROR 模型

基于 ROR 模型的形式化安全分析是一种强大的认证协议安全证明方法[6,19]。为了额外描述敌手的雾节点泄露攻击和云服务器泄露攻击，本节扩展了 ROR 模型，扩展后该模型的组成部分如下。

1）参与者。协议的参与者包括物联网医疗设备（MD_i）、雾节点（FN_j）和云服务器（CS_k），这 3 类参与者的第 u,t,s 个实例可分别表示为 $\Pi_{MD_i}^u$，$\Pi_{FN_j}^t$，$\Pi_{CS_k}^s$，也称为预言机。

2）接受状态。当一个实例 Π_X^n 接收到最后一次交换的消息，就会转为接受状态。如果将所有交换的消息顺序连接，就构成了当前会话 Π_X^n 的会话标识 sid。

3）伙伴关系。如果同时满足以下 3 个约束条件，则两个实例 $\Pi_X^{n_1}$ 和 $\Pi_X^{n_2}$ 是伙伴关系：(1) $\Pi_X^{n_1}$ 和 $\Pi_X^{n_2}$ 都处于接受状态；(2) $\Pi_X^{n_1}$ 和 $\Pi_X^{n_2}$ 相互认证，且具有相同的会话标识 sid；(3) $\Pi_X^{n_1}$ 和 $\Pi_X^{n_2}$ 互为伙伴关系。

4）新鲜度。如果两个参与者之间建立的会话密钥没有通过下面定义的 $Reveal$ 查询泄露给敌手 $\mathcal{A}$，就称这两个参与者的实例 $\Pi_X^{n_1}$ 和 $\Pi_X^{n_2}$ 是新鲜的。

5）敌手。在扩展 ROR 模型中，一个敌手 $\mathcal{A}$ 可以完全控制通信信道，也就是说，$\mathcal{A}$ 不仅可以拦截（窃听）、修改和删除传输的消息，还可以制造新信息并注入通信信道中。此外，$\mathcal{A}$ 还可以对实例执行以下查询。

(1) $Execute(\Pi_{MD_i}^u,\Pi_{FN_j}^t,\Pi_{CS_k}^s)$：该查询模拟敌手 $\mathcal{A}$ 的被动（窃听）攻击。在这个查询中，$\mathcal{A}$ 可以拦截这 3 个参与者之间通过公开通信信道传输的所有消息。

(2)$Send(\Pi_X^n, m)$：该查询模拟敌手$\mathcal{A}$的主动攻击。$\mathcal{A}$通过这个查询，可以向实例Π_X^n发送一个消息m，并能收到其响应的消息。

(3)$Reveal(\Pi_X^n)$：该查询模拟会话密钥泄露攻击。执行这样的查询后，$\mathcal{A}$就可以获得实例Π_X^n及其伙伴当前生成的会话密钥。

(4)$CorruptMD(\Pi_{MD_i}^u)$：该查询模拟物联网医疗设备被盗攻击。通过使用这个查询，$\mathcal{A}$可以获取到存储在医疗设备中所有的凭证信息。

(5)$CorruptFN(\Pi_{FN_j}^r)$：该查询模拟雾节点捕获攻击。执行这个查询后，雾节点中存储的所有秘密凭证信息都将被透露给敌手$\mathcal{A}$。

(6)$CorruptCS(\Pi_{CS_k}^s)$：该查询模拟云服务器捕获攻击。$\mathcal{A}$通过执行此查询，提取出云服务器中所存储的凭证信息。

(7)$Test(\Pi_X^n)$：该查询用于模拟实例建立的会话密钥的语义安全性。在此$Test$查询下，输出的结果由一个随机位c决定，c的取值为0或1的概率都是1/2。当敌手$\mathcal{A}$执行$Test$查询时，如果实例的会话钥已建立且是新鲜的，则$c=1$时，返回真实的会话钥；而$c=0$时，返回一个与会话钥相同位数的随机数。如果还未建立会话钥，则输出为未定义的符号$\perp$。

6)会话密钥的语义安全性。在扩展ROR模型中，敌手$\mathcal{A}$需要区分出返回的实例真实会话钥与等长的随机数。$\mathcal{A}$可以通过多次执行上述$Execute$，$Send$，$Reveal$，$Corrupt$以及$Test$等查询来进行区分。游戏结束后，$\mathcal{A}$将给出猜测的随机数c'，如果$c'=c$成立，就称$\mathcal{A}$赢得了游戏。如果用$Succ$表示$\mathcal{A}$赢得游戏的事件，那么A破坏协议$\mathcal{P}$的会话钥语义安全性的优势就可以定义为$Adv_{\mathcal{P}}(\mathcal{A})=|2\cdot Pr[Succ]-1|$。如果存在一个大于零且无限趋近于零的数，且对任意概率多项式时间内的$\mathcal{A}$，始终满足$Adv_{\mathcal{P}}(\mathcal{A})\leqslant\epsilon$，那么就称协议$\mathcal{P}$具有ROR模型下的会话钥语义安全性。

7)随机预言机。敌手$\mathcal{A}$和其他参与协议的通信实体都可以访问一个抗碰撞的单向加密哈希函数$h(\cdot)$，该函数被模拟为一个随机预言机。

6.2 使用扩展ROR模型的形式化安全证明

此节将分别证明所提协议中的医疗设备MD_i与雾节点FN_j之间，以及雾节点FN_j与云服务器CS_k之间认证过程的密钥语义安全性。

定理1 令$\mathcal{P}_1$表示医疗设备MD_i与雾节点FN_j之间的认证过程。在扩展ROR模型中，敌手$\mathcal{A}$在概率多项式时间t内破坏协议$\mathcal{P}_1$的密钥语义安全性的优势为：

$$Adv_{\mathcal{P}_1}(\mathcal{A})\leqslant\frac{q_h^2}{|Hash|}+2Adv_{\mathcal{P}_1}^{\mathrm{ECCDHP}}(\mathcal{A})$$

其中，q_h表示哈希查询的次数，$|Hash|$表示抗碰撞的单向加密哈希函数$h(\cdot)$的范围空间，$Adv_{\mathcal{P}_1}^{\mathrm{ECCDHP}}(\mathcal{A})$表示$\mathcal{A}$解决椭圆曲线判定性Diffie-Hellman问题(ECDDHP)的优势。

证明 与文献[45-46]中的证明相似，在此证明中，定义了包括$G_i(i=0,1,2,3,4)$在内的5个游戏，用于证明$\mathcal{P}_1$的语义安全性。其中，在游戏G_i中成功猜出随机位c的事件被定义为$Succ_i$。详细证明如下。

G_0：在此游戏中，模拟敌手$\mathcal{A}$对$\mathcal{P}_1$过程的真实攻击。由于在游戏开始之前，$\mathcal{A}$并未被赋予任何额外的能力，因此这时$\mathcal{A}$猜测出正确c的优势为：

$$Adv_{\mathcal{P}_1}(\mathcal{A})=|2\cdot Pr[Succ_0]-1| \tag{1}$$

G_1：此游戏模拟敌手$\mathcal{A}$的被动攻击。$\mathcal{A}$可以执行$Execute(\Pi_{MD_i}^u,\Pi_{FN_j}^r,\Pi_{CS_k}^s)$查询，在$\mathcal{P}_1$中拦截$MD_i$和$FN_j$在公共信道上传输的所有消息，包括消息$M_{MF_1}=\{TID_{MD_i}^{new},V_1,T_{MF_1}\}$，$M_{MF_3}=\{R_{MD_i},TID_{MD_i}^*,SKV,T_{MF_3}\}$以及$FN_j$发送给$MD_i$的消息$M_{MF_2}=\{R_{FN_j},V_2,T_{MF_2}\}$。在游戏完成后，$\mathcal{A}$还将执行$Test(\Pi_X^n)$查询，用于确定输出结果是真实会话钥还是与会话钥等长的随机数。$\mathcal{P}_1$中MD_i和FN_j的会话钥为$SK_{ij}=h(S_{MD_i}\cdot R_{FN_j}\parallel T_{MF_3})$，其中$S_{MD_i}=h(r_{MD_i}\parallel PID_{MD_i}\parallel T_{MF_1})$。而$\mathcal{A}$无法通过执行$Execute$查询获取其中的秘密参数$r_{MD_i}$和$PID_{MD_i}$，进而也无法计算出这个会话密钥。因此$\mathcal{A}$赢得游戏的优势并未增加，即：

$$|Pr[Succ_1]-Pr[Succ_0]|=0 \tag{2}$$

G_2：在这个游戏中，增加了$Send$查询和哈希查询，用于模拟敌手$\mathcal{A}$的主动攻击。$\mathcal{A}$将首先伪造一条消息并通过$Send(\Pi_X^n,m)$查询发送给参与者，并被允许多次使用哈希查询来检验哈希碰撞。然而，参与者交换的消息中包含临时随机数、时间戳等随时间变化的信息，并且$\mathcal{P}_1$采用的哈希函数是抗碰撞的，因此$\mathcal{A}$无法通过$Send$查询寻找到哈希碰撞的情况。根据生日悖论原理[44]，可知：

$$|Pr[Succ_2]-Pr[Succ_1]|\leqslant\frac{q_h^2}{2|Hash|} \tag{3}$$

G_3：在游戏G_3中模拟敌手$\mathcal{A}$的医疗设备捕获攻击和雾节点泄露攻击，为其增加$CorruptMD(\Pi_{MD_i}^u)$和$CorruptFN(\Pi_{FN_j}^r)$查询。通过执行上述查询，可以分别获取到医疗设备中存储的$\{TID_{MD_i}^{old},TID_{MD_i}^{new},PID_{MD_i},k_{MD_i}\}$和雾节点存储的$\{TID_{FN_j}^{old},TID_{FN_j}^{new},PID_{FN_j},f(PID_{FN_j},y),k_{FN_j}\}$以及$\{(TID_{MD_i},PID_{MD_i})|i=1,2,\cdots,n_{MD}\}$。由于$\mathcal{A}$不知道临时随机数和时间戳的值，因此无法直接计算会话钥$SK_{ij}=h(h(r_{MD_i}\parallel PID_{MD_i}\parallel T_{MF_1})\cdot h(r_{FN_j}\parallel PID_{FN_j}\parallel T_{MF_2})\cdot P\parallel T_{MF_3})$，也就是说，$\mathcal{A}$在$G_3$中也并未增加任何优势，所以有：

$$|Pr[Succ_3]-Pr[Succ_2]|=0 \tag{4}$$

G_4：最后一个游戏将执行一种额外的主动攻击，该攻击用于模拟敌手$\mathcal{A}$对ECCDPH问题的破坏。已知通过执行上述$G_i(i=0,1,2,3,4)$，$\mathcal{A}$还是无法直接计算出会话钥。然而，通过游戏G_1，$\mathcal{A}$已知R_{MD_i}，R_{FN_j}和P，而$SK_{ij}=S_{MD_i}\cdot R_{FN_j}=S_{MD_i}\cdot S_{FN_j}\cdot P$。也就是说，$\mathcal{A}$能够以在概率多项式时间内解决ECCDPH问题的优势得到会话密钥SK_{MD_i,FN_j}的值，故有：

$$|Pr[Succ_4]-Pr[Succ_3]|\leqslant Adv_{\mathcal{P}_1}^{\mathrm{ECCDHP}}(A) \tag{5}$$

至此，$\mathcal{A}$已经执行完了所有查询。$\mathcal{A}$只能使用$Test(\Pi_X^n)$查询试图猜测c的值，故：

$$Pr[Succ_4]=\frac{1}{2} \tag{6}$$

根据式(1)－式(6)，可以得出：

$$Adv_{\mathcal{P}_1}(\mathcal{A})\leqslant\frac{q_h^2}{|Hash|}+2Adv_{\mathcal{P}_1}^{\mathrm{ECCDHP}}(\mathcal{A})$$

证毕.

定理2 令$\mathcal{P}_2$表示雾节点FN_j与云服务器CS_k之间的认证过程。在扩展ROR模型中，敌手$\mathcal{A}$在概率多项式时间t内破坏$\mathcal{P}_2$密钥语义安全性的优势为：

$$Adv_{\mathcal{P}_2}(\mathcal{A})\leqslant\frac{q_h^2}{|Hash|}$$

其中，q_h表示哈希查询的次数，$|Hash|$表示抗碰撞的单向加密哈希函数$h(\cdot)$的范围空间。

证明　与定理1的证明类似，定义$G_i'(i=0,1,2,3)$这4个游戏，在游戏G_i'中成功猜出随机位c的事件被定义为$Succ_i'$。详细证明如下。

G_0'：与G_0阶段相同，$\mathcal{A}$并未被赋予任何额外的能力。这时$\mathcal{A}$猜测出正确的c的优势为：

$$Adv_{\mathcal{P}_2}(\mathcal{A})=|2\cdot Pr[Succ_0']-1| \tag{7}$$

G_1'：此游戏模拟敌手$\mathcal{A}$的被动攻击。$\mathcal{A}$可以执行$Execute(\Pi_{MD_i}^u,\Pi_{FN_j}^s,\Pi_{CS_k}^s)$查询，拦截$FN_j$和$CS_k$在公共信道传输的所有消息，即$M_{FC_1}=\{TID_{FN_j}^{new},A,T_{FC_1}\}$和$M_{FC_2}=\{B,C,D,SKV_2,T_{FC_2}\}$。在游戏完成后，$\mathcal{A}$执行$Test(\Pi_X^n)$查询，用于确定输出结果是真实会话钥还是等长的随机数。而$SK_{kj}=h(H_1\parallel H_2\parallel f(PID_{CS_k},PID_{FN_j}))$，其中，$H_1=h(r_{FN_j}\parallel k_{FN_j}\parallel PID_{FN_j}\parallel T_{FC_1})$，$H_2=h(r_{CS_k}\parallel k_{CS_k}\parallel T_{FC_2})$。而$\mathcal{A}$在$Execute$查询中无法获取$r_{FN_j}$，$r_{CS_k}$，$k_{FN_j}$，$k_{CS_k}$，$PID_{FN_j}$和$f(PID_{CS_k},PID_{FN_j})$，也就无法计算这个会话密钥。因此$\mathcal{A}$赢得游戏的优势并未增加，即：

$$|Pr[Succ_1']-Pr[Succ_0']|=0 \tag{8}$$

G_2'：与G_2类似，在增加$Send(\Pi_X^n,m)$查询和哈希查询的情况下，由于传递的消息中包含临时随机数、时间戳等随时间变化的信息，且$\mathcal{P}_2$采用的哈希函数也是抗碰撞的，根据生日悖论原理，可知：

$$|Pr[Succ_2']-Pr[Succ_1']|\leqslant\frac{q_h^2}{2|Hash|} \tag{9}$$

G_3'：此游戏模拟云服务器泄露攻击，增加$Corrupt\text{-}CS(\Pi_{CS_k}^s)$查询。$\mathcal{A}$执行此查询可以获取$CS_k$中存储的$\{PID_{CS_k},f(PID_{CS_k},y),k_{CS_k},Pub_{CS_k}\}$和$\{(TID_{FN_j}^{old},TID_{FN_j}^{new},PID_{FN_j})|j=1,2,\cdots,n_{FN}\}$。由于$\mathcal{A}$不知道随机数、时间戳、$k_{FN_j}$和$PID_{FN_j}$，就无法计算$SK_{kj}=h(H_1\parallel H_2\parallel f(PID_{CS_k},PID_{FN_j}))$，其中，$H_1=h(r_{FN_j}\parallel k_{FN_j}\parallel PID_{FN_j}\parallel T_{FC_1})$，$H_2=h(r_{CS_k}\parallel k_{CS_k}\parallel T_{FC_2})$，因此：

$$|Pr[Succ_3']-Pr[Succ_2']|=0 \tag{10}$$

至此，$\mathcal{A}$已经执行完了所有查询。$\mathcal{A}$只能使用$Test(\Pi_X^n)$查询试图猜测c的值，故有：

$$Pr[Succ_3']=\frac{1}{2} \tag{11}$$

根据式(7)－式(11)，可以得出：

$$Adv_{\mathcal{P}_2}(\mathcal{A})\leqslant\frac{q_h^2}{|Hash|}$$　　证毕.

6.3　基于评价标准的启发式安全分析

本节将基于4.3节中定义的评价标准，对所提出的协议进行分析，以证明其满足4.3节中列出的所有安全属性。

1)相互认证和密钥协商(EC1，EC2)

如5.3节所述，本协议成功实现了医疗设备与雾节点之间，以及雾节点和云服务器之间的相互认证和密钥建立。

2)匿名性和不可追溯性(EC3)

在提出的协议中，所有实体都使用了伪身份，例如RA的伪身份$PID_{RA}=h(ID_{RA}\parallel s)$，其中$ID_{RA}$为RA的真实身份，$s$为系统私钥。由于哈希函数的单向性，通过伪身份是无法计算出真实身份的，因此在消息传输过程中并没有真实身份的泄露。此外，除RA外的其余实体还采用了临时身份，临时身份初始是在注册阶段由RA随机选择，在每次会话完成后都会进行更新。而会话过程中所传递的消息中的时间戳、公共参数、签名等各种信息都是动态变化的，因此，协议实现了匿名和不可追踪性。

3)密钥前向/后向安全性(EC4)

对于FN_j与MD_i共享的会话密钥$SK_{ji}=h(h(r_{FN_j}\parallel PID_{FN_j}\parallel T_{MF_2})\cdot h(r_{MD_i}\parallel PID_{MD_i}\parallel T_{MF_1})\cdot P)$，其中包含的随机数$r$和时间戳$T$是随会话变化而动态变化的。$FN_j$与$CS_k$之间的会话密钥同理。因此，即使敌手获取了当前实例间的会话密钥，也无法推测出它们之前或之后会话的会话密钥。所以协议保障了密钥前向/后向安全性。

4)抗去同步攻击(EC5-1)

协议在医疗设备和雾节点中存储TID^{old}和TID^{new}两个临时身份信息，在每次通信中都会分别更新对应实体的这两个值。当出现敌手通过拦截公开信道消息导致某次会话失效，而只有其中一方更新了临时身份的情况时，实体仍可以通过TID^{old}还原TID^{new}，使下一次会话不受影响，即协议可以抵抗去同步攻击。

5)抗重放攻击(EC5-2)

协议参与者在通信过程中传输的消息中，都附加了时间戳或随机参数信息。例如，消息$M_{MF_1}=\{TID_{MD_i}^{new},V_1,T_{MF_1}\}$中，$T_{MF_1}$是时间戳；消息$M_{MF_3}=\{R_{MD_i},TID_{MD_i}^*,SKV,T_{MF_3}\}$中，$T_{MF_3}$为时间戳，$R_{MD_i}$，$TID_{MD_i}^*$和$SKV$的计算过程包含随机数和时间戳。会话改变时，时间戳和随机参数都会变化。因此，协议能够抵抗重放攻击。

6)抗中间人攻击(EC5-3)

假设一个敌手$\mathcal{A}$从开放通道拦截了一个医疗设备MD_i的请求信息M_{MF_1}，并试图发起另一个有效的请求信息，以使FN_j无法检测篡改信息的真实性。敌手$\mathcal{A}$可以自己生成时间戳$T_{MF_1}^{\mathcal{A}}$，但由于不知道MD_i预加载凭证中的信息，例如PID_{MD_i}，TID_{MD_i}和C_i，也无法计算$R_i=PUF(C_i)$，因此无法构建出合法的M_{MF_1}以通过后续FN_j的认证。同样地，$\mathcal{A}$也无法构建其他有效的消息来代替实际截获的消息。因此，协议对于中间人攻击具有抵抗性。

7)抗假冒攻击(EC5-4)

(1)医疗设备假冒攻击

假设一个敌手$\mathcal{A}$试图假冒一个合法的智能医疗设备MD_i以完成与雾节点FN_j的认证，那么$\mathcal{A}$需要构建消息，如$M_{MF_1}=\{TID_{MD_i}^{new},V_1,T_{MF_1}\}$。敌手可以自己生成时间戳$T_{MF_1}^{\mathcal{A}}$。然而，$\mathcal{A}$还需要计算出$V_1^{\mathcal{A}}=h(PID_{MD_i}\parallel C_i\parallel R_i\parallel T_{MF_1}^{\mathcal{A}})$。为了使该假冒攻击可行，$\mathcal{A}$需要证明$V_1^{\mathcal{A}}=V_1$。然而，$\mathcal{A}$并不知道$PID_{MD_i}$，$C_i$和$R_i$的值，因此难以计算正确的$V_1^{\mathcal{A}}$。同时，$\mathcal{A}$尽管可以窃听某个会话中$TID_{MD_i}^{new}$的值，但该值会在当前会话结束后更新。类似地，因为缺少PID_{MD_i}，$\mathcal{A}$无法计算出R_{MD_i}，SK_{ij}，TID'_{MD_i}，SKV等值，进而无法构建出$M_{MF_3}^{\mathcal{A}}=$

M_{MF_3}。因此,协议可以抵抗医疗设备假冒攻击。

(2)雾节点假冒攻击

与医疗设备假冒攻击一节类似,由于敌手不知道参数 PID_{MD_i},R_i,PID_{FN_j},k_{FN_j} 和 TID_{FN_j} 的值,因此无法构建出与 MD_i 认证所需的 $M^{\mathcal{A}}_{MF_2}=M_{MF_2}$ 和与 CS_k 沟通的 $M^{\mathcal{A}}_{FC_1}=M_{FC_1}$。故敌手无法实施雾节点假冒攻击。

(3)云服务器假冒攻击

与上述攻击类似,敌手 $\mathcal{A}$ 若试图假冒云服务器,则需要构建出与 FN_j 通信的认证信息 $M^{\mathcal{A}}_{FC_2}=M_{FC_2}$,尽管 $\mathcal{A}$ 可以通过收到的 M_{FC_1} 得知 TID_{FN_j},但由于无法得到 PID_{FN_j},因而无法还原 H_1。同时,由于 $\mathcal{A}$ 并不知道 k_{CS_k} 和 PID_{CS_k} 的值,无法计算 H_2 和正确的共享密钥,因此更无法构造出正确的 $M^{\mathcal{A}}_{FC_2}=M_{FC_2}$。所以,协议对于云服务器假冒攻击也具有抵抗性。

8)抗特权内幕攻击(EC5-5)

如 5.2 节注册阶段所述,除 RA 外的所有实体的注册阶段都是在安全环境下实现的,且在注册阶段中,需要注册的实体都不会发送注册信息给 RA,而仅由 RA 为注册实体生成并提前加载所有凭证。凭证中加载的相关信息是伪身份和临时身份,RA 也并不存储用户真实身份信息。因此,RA 的特权内部用户也无法获取更多的信息。故协议可以抵抗特权内幕攻击。

9)抗短暂秘密泄露攻击(EC5-6)

短暂秘密泄露攻击是指 $\mathcal{A}$ 获取了协议中一个实体 B 的长期或短期秘密,并试图冒充另一个实体与 B 通信。以 $SK_{ij}=h(S_{MD_i}\cdot R_{FN_j}\parallel R_i\parallel T_{MF_3})$ 为例,其中 $S_{MD_i}=h(r_{MD_i}\parallel PID_{MD_i}\parallel T_{MF_1})$,$R_{FN_j}=h(r_{FN_j}\parallel PID_{FN_j}\parallel T_{MF_2})\cdot P$。可以看出,加密过程中除了包含一些长期秘密(如伪身份等),还使用了短期秘密,如随机数、时间戳等信息,因此敌手依旧不能仅通过掌握长期秘密或短期秘密来建立与另一个实体的会话密钥。同理,$\mathcal{P}_2$ 过程也拥有对短暂秘密泄露攻击的抵抗性。

10)抗医疗设备捕获攻击(EC6)

协议可以抵御医疗设备捕获攻击。假设敌手获取了医疗设备 MD_i 中存储的全部信息,即秘密凭证 $\{TID^{old}_{MD_i},TID^{new}_{MD_i},PID_{MD_i},C_i\}$。而要想构造会话钥 $SK_{ij}=h(S_{MD_i}\cdot R_{FN_j}\parallel R_i\parallel T_{MF_3})$,敌手必须同时计算出 S_{MD_i},R_{FN_j} 和 R_i。由于敌手并不知道 r_{MD_i},r_{FN_j},PID_{FN_j},T_{MF_1},T_{MF_2},T_{MF_3} 这些参数,也无法直接拿到 MD_i 来计算 R_i,因此无法计算会话钥 SK_{MD_i,FN_j}。

11)抗雾节点泄露攻击(EC7)

假设雾节点 FN_j 中存储的秘密凭证 $\{TID^{old}_{FN_j},TID^{new}_{FN_j},PID_{FN_j},f(PID_{FN_j},y),k_{FN_j}\}$ 和 $\{(TID_{MD_i},PID_{MD_i})\mid i=1,2,\cdots,n_{MD}\}$ 被泄露,但想要计算 SK_{ji},敌手还缺乏 r_{FN_j},r_{MD_i},PID_{MD_i}(敌手并不知道是哪个 TID_{MD_i} 对应的 PID_{MD_i}),R_i,T_{MF_1},T_{MF_2},T_{MF_3} 这些参数。

而对于 $SK_{jk}=h(H_1\parallel H_2\parallel f(PID_{FN_j},PID_{CS_k}))$,其中 $H_1=h(r_{FN_j}\parallel k_{FN_j}\parallel PID_{FN_j}\parallel T_{FC_1})$,$H_2=h(r_{CS_k}\parallel k_{CS_k}\parallel T_{FC_2})$,由于敌手无法获取 r_{FN_j},r_{CS_k},k_{CS_k},PID_{CS_k},T_{FC_1} 和 T_{FC_2},因此也无法得到 SK_{jk}。故协议可以抵抗雾节点泄露攻击。

12)抗云服务器泄露攻击(EC8)

假设云服务器被泄露,敌手可以获取到 $\{PID_{CS_k},f(PID_{CS_k},y),k_{CS_k},Pub_{CS_k}\}$ 和 FN_j 参数 $\{(TID^{old}_{FN_j},TID^{new}_{FN_j},PID_{FN_j})\mid j=1,2,\cdots,n_{FN}\}$。然而,密钥 $SK_{kj}=h(H_1\parallel H_2\parallel f(PID_{CS_k},PID_{FN_j}))$ 的计算缺乏关键参数 r_{FN_j},r_{CS_k},k_{FN_j},PID_{FN_j},T_{FC_1} 和 T_{FC_2}。因此,协议可以抵御云服务器泄露攻击。

6.4 基于 AVISPA 工具的实验仿真分析

本节将使用 AVISPA[47] 协议自动化安全分析工具,验证提出的协议对窃听、中间人等攻击的抵抗性和会话密钥的安全性。

首先,使用 HLPSL 语言定义协议中涉及的 4 个基本角色(Basic Role),分别是注册权威 RA、医疗设备 MD、雾节点 FN 和云服务器 CS。RA 在接收到开始信号 start 后,由初始状态 0 转化为状态 1,启动系统初始化并分别执行 MD,FN 和 CS 的注册过程。

在 MD 和 FN 认证过程中,首先,MD 在接收到 RA 传来的注册信息后由状态 0 转为状态 1,计算并发送 M_{MF_1}。然后在收到 M_{MF_2} 后转为状态 2,计算出与 FN 的会话密钥并发送 M_{MF_3}。FN 也同样在收到注册信息时转为状态 1,在收到 M_{MF_1} 后跳转到状态 2 并发送 M_{MF_2},在收到 M_{MF_3} 后转为状态 3,计算并验证会话密钥,实现相互认证。而对于 FN 和 CS 的认证过程,CS 在注册完成后变为状态 1,收到 M_{FC_1} 后转为状态 2,并发送 M_{FC_2}。FN 注册完成后从状态 0 变为状态 1,发送 M_{FC_1}。收到 M_{FC_2} 后,转为状态 2,计算出与 CS 的会话密钥和会话密钥验证器,完成认证过程。

除了上述基本角色的规范外,还通过 HLPSL 语言[48] 定义了会话、环境和目标的规范。其中,会话是对基本角色的实例化,环境是整个代码执行的起点,包含全局变量、入侵者知识和多个会话,而目标给出了协议需要实现的保密目标和认证目标。

图 3 中展示了协议在 SPAN+AVISPA 上的仿真实验结果,其中上方为 MD 和 FN 认证过程的结果,下方为 FN 和 CS 认证过程部分的结果,左侧为 OFMC 后端,右侧为 CL-AtSe 后端。实验结果表明,协议在 OFMC 和 CL-AtSe 后端均是安全的,能够成功抵抗在 Dolev-Yao 威胁模型下的重放攻击、中间人攻击和窃听攻击,实现了相互认证与会话密钥的安全性。

```
SUMMARY
  SAFE

DETAILS
  BOUNDED NUMBER OF SESSIONS
  TYPED MODEL

PROTOCOL
  /home/span/span/testsuite/results/FCAKA.if

GOAL
  As Specified

BACKEND
  CL-AtSe
```

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED NUMBER OF SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/FCAKA.if
GOAL
  as specified
BACKEND
  OFMC
COMMENTS
```

```
SUMMARY
  SAFE

DETAILS
  BOUNDED NUMBER OF SESSIONS
  TYPED MODEL

PROTOCOL
  /home/span/span/testsuite/results/MFAKA.if

GOAL
  As Specified

BACKEND
  CL-AtSe
```

```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED NUMBER OF SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/MFAKA.if
GOAL
  as specified
BACKEND
  OFMC
COMMENTS
```

图 3 SPAN+AVISPA 仿真实验结果

Fig. 3 Result of SPAN+AVISPA simulation experiment

7　性能分析

本章详细比较了提出的协议与 Fan 等[49]、Hewa 等[50]、Huang 等[51]以及 Li 等[25]协议的功能和安全属性、通信成本以及计算成本。

7.1　功能和安全属性

将 Fan 等[49]、Hewa 等[50]、Huang 等[51]以及 Li 等[25]提出的协议与第 6 章所证明的本文协议的功能和安全属性进行汇总，表 3 列出了 5 个协议在功能和安全属性方面的比较情况。其中：Fan 等[49]的协议缺乏不可追溯性，容易受到雾节点泄露攻击，并且不满足确定的安全属性；Hewa 等[50]的协议无法抵抗密钥前向/后向安全性和云服务器泄露攻击；Huang 等[51]的协议未能抵抗智能设备捕获攻击和特权内幕攻击；Li 等[25]的协议无法抵御雾节点泄露和云服务器泄露攻击；本文协议支持表 3 中列出的所有功能和安全属性。

表 3　功能和安全属性比较

Table 3　Comparison of performance and security characteristics

功能/安全属性	文献[49]	文献[50]	文献[51]	文献[25]	本文协议
F_1	×	√	√	√	√
F_2	√	×	√	√	√
F_3	√	√	√	√	√
F_4	√	√	√	√	√
F_5	√	√	√	√	√
F_6	√	√	√	√	√
F_7	√	√	√	√	√
F_8	—	√	√	√	√
F_9	—	√	×	√	√
F_{10}	×	√	√	×	√
F_{11}	—	×	√	×	√
F_{12}	—	√	×	√	√
F_{13}	—	√	√	√	√
F_{14}	√	√	√	√	√
F_{15}	√	√	√	√	√

注：F_1 为匿名性和不可追溯性；F_2 为密钥前向/后向安全性；F_3 为抗去同步攻击；F_4 为抗重放攻击；F_5 为抗中间人攻击；F_6 为抗物联网设备假冒攻击；F_7 为抗雾节点假冒攻击；F_8 为抗云服务器假冒攻击；F_9 为抗物联网设备捕获攻击/智能卡被盗攻击；F_{10} 为抗雾节点泄露攻击；F_{11} 为抗云服务器泄露攻击；F_{12} 为抗特权内幕攻击；F_{13} 为抗短暂秘密泄露攻击；F_{14} 为相互认证；F_{15} 为密钥协商；√ 表示支持此属性；× 表示不支持此属性；— 表示不适用。

7.2　通信成本

为了公平地比较通信成本，本节统一规定协议中所涉及的数据大小。假设 160 位的椭圆曲线与 1024 位的 RSA 公钥密码系统提供了相同的安全级别，ECC 上的点 P 为(160+160)=320 位(P 的 x 和 y 坐标分别是 160 位)，临时随机数由 $GF(q)$ 生成，也是 160 位。假设身份、伪身份、临时身份的长度均为 160 位，对称加密/解密操作的长度为 160 位，时间戳为 32 位，哈希摘要的长度为 256 位(使用 SHA-256 算法)。

在本文协议中，$\mathcal{P}_1$ 过程交换的消息 $M_{MF_1}=\{TID_{MD_i}^{new}, V_1, T_{MF_1}\}$，$M_{MF_2}=\{R_{FN_j}, V_2, T_{MF_2}\}$ 和 $M_{MF_3}=\{R_{MD_i}, TID_{MD_i}^{*}, SKV, T_{MF_3}\}$ 分别需要(160+256+32)=448 位、(320+256+32)=608 位、(320+256+256+32)=864 位，总成本为 1920 位。$\mathcal{P}_2$ 交换的两条消息长度分别为(160+256+32)=448 位、(256+256+256+256+32)=1056 位，总成本为 1504 位。整个协议的通信总成本为 3424 位。

Fan 等[49]的协议涉及 5 条消息的交换，大小分别为 832 位、992 位、832 位、832 位和 320 位，总成本为 3808 位。

Hewa 等[50]的协议共传递了 4 条消息，大小分别为 960 位、320 位、416 位和 1120 位，总成本为(960+320+416+1120)=2816 位。

Huang 等[51]的协议包含 4 条消息的交换，大小分别为 928 位、608 位、864 位和 1088 位，总通信成本为 3488 位。

Li 等[25]的协议中共发送了 4 条消息，消息大小分别为 1184 位、2688 位、1504 位和 1248 位，总成本为(1184+2688+1504+1248)=6624 位。

图 4 展示了各协议通信成本的比较情况。

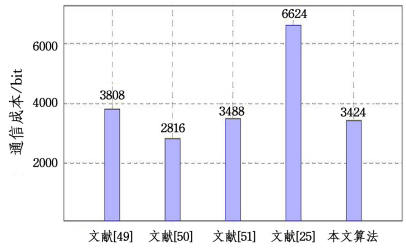


图 4　通信成本比较

Fig. 4　Comparison of communication costs

7.3　计算成本

对于计算成本的比较，本节定义单向加密哈希函数、ECC 点加、ECC 点乘、双线性配对、模乘法、t 度多项式计算、PUF 操作和对称加密/解密操作所需的时间分别为 T_h，T_{epa}，T_{epm}，T_{bp}，T_m，T_{poly}，T_p 和 T_s。t 度多项式的计算需要进行 t 次模乘法和 t 次模加法，模加法相比模乘法运算的计算时间可以忽略不计，所以有 $T_{poly}=tT_m+tT_a\approx tT_m$($T_a$ 表示模加法)。根据文献[2]、文献[6]和文献[13]，这些密码原语的近似运行时间如表 4 所列。

表 4　密码原语的近似运行时间

Table 4　Approximate runtime of password primitives

(ms)

实体	T_h	T_{epa}	T_{epm}	T_{bp}	T_m	T_p	T_s
物联网设备	0.056	0.081	13.405	32.713	0.008	0.023	0.224
服务器	0.007	0.013	2.165	5.427	0.001	—	0.028

本文协议在 $\mathcal{P}_1$ 过程中，医疗设备所需计算时间为 $7T_h+3T_{epm}+T_m+T_p+T_{epa}\approx 40.719$ ms，雾节点计算时间为 $6T_h+2T_{epm}+T_m\approx 4.373$ ms，而 $\mathcal{P}_2$ 中雾节点和云服务器的计算成本都为 $7T_h+T_{poly}\approx(0.049+0.001t)$ ms。由于 t 的值应该不小于 n_{FN} 和 n_{CS} 中的最大值，即使 t 取值为 1000，$\mathcal{P}_2$ 的服务器总计算成本也仅为 2.098 ms。整个协议的总计算成本为 47.19 ms。

Fan 等[49]协议的计算成本为物联网设备端 $3T_h+3T_{epm}+2T_{bp}+T_m+T_s\approx 106.041$ ms 和服务器端 $3T_h+3T_{epm}+2T_{bp}+T_m+T_s\approx 17.399$ ms，总成本为 123.44 ms。

为了保证比较的公平性，本文忽略了 Hewa 等[50]协议中智能合约的计算成本，得到协议物联网节点所需的计算成本为 $8T_h+8T_{epm}+5T_s\approx 108.808$ ms，而雾节点和云服务器端计算成本为 $6T_h+6T_{epm}+6T_m+3T_s\approx 13.122$ ms，总成本为 121.93 ms。

Huang 等[51]的协议中，物联网设备端计算成本为 $5T_h+$

$5T_{epm}+T_s\approx 67.529$ ms，服务器端成本为 $18T_h+12T_{epm}+3T_s\approx 26.19$ ms，总成本为 93.719 ms。

Li 等[25]的协议中用户设备和服务器的计算成本分别为 $7T_h+4T_{epm}\approx 54.012$ ms 和 $17T_h+11T_{epm}\approx 23.934$ ms，总计算成本为 77.946 ms。

本文协议与其他协议的计算成本的比较情况如图 5 所示。结果表明，所提出的协议的计算成本相对较小，且相比其他协议在服务器端更具优势。

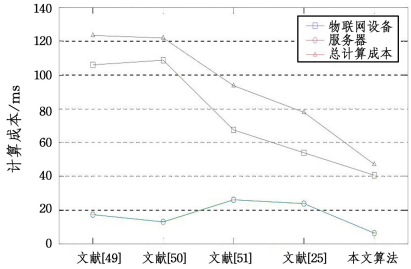


图 5 计算成本比较

Fig. 5 Comparison of computation costs

7.4 能源消耗

根据文献[52]和文献[53]，一些密码学原语在 133 MHz 的“StrongARM”CPU 上运行所需的能源消耗情况如表 5 所列。

表 5 密码学原语的能源消耗

Table 5 Energy consumption of cryptographic primitives

密码学原语	能源消耗/mj
SHA-1 哈希操作	0.000108
对称加密/解密	0.00217
双线性配对	47
ECC 点乘法	8.8
ECC 点加法	0.001085

由于文献[52]和文献[53]中并未给出 SHA-256 操作的能源消耗，因此统一按照 SHA-1 操作来计算。根据 7.3 节中各原语的使用次数，可以得出各协议能源消耗近似比较情况，如表 6 所列。

表 6 能源消耗近似比较结果

Table 6 Approximate comparison results of energy consumption

方法	能源消耗/mj
Fan 等[49]	240.804988
Hewa 等[50]	123.218872
Huang 等[51]	149.611164
Li 等[25]	132.002592
本文协议	44.004001

7.5 未知攻击下的性能

本节将比较各协议面对未知攻击时的性能。虽然已经证明了提出的协议在各种已知攻击下具备鲁棒的安全性，然而，还需要考虑一些未曾预计的未知攻击对协议的影响。根据文献[54]和文献[55]，分别定义协议遭遇未知攻击下的平均通信成本C_{Avg}和协议遭遇未知攻击后执行失败的通信成本C_{Fail}，如式(12)、式(13)所示：

$$C_{Avg}=\frac{C_{Succ}\times(1-P)+C_{Fail}\times P}{1-P} \tag{12}$$

$$C_{Fail}=\sum_{n=1}^{Num}C_n\times\frac{1}{Num} \tag{13}$$

其中，C_{Succ}代表协议在未知攻击下执行成果的通信成本，P 代表协议遭受未知攻击的概率，C_n代表在步骤 n 发生未知攻击时的通信成本，Num 代表协议认证过程中传递的消息数，在步骤 n 发生未知攻击的概率为$\frac{1}{Num}$。

表 7 未知攻击下的通信开销

Table 7 Communication overhead under unknown attack

方法	消息数	总通信成本
文献[49]	5	$\frac{19040\times(1-P)+12608\times P}{5\times(1-P)}$
文献[50]	4	$\frac{11264\times(1-P)+6752\times P}{4\times(1-P)}$
文献[51]	4	$\frac{13952\times(1-P)+8352\times P}{4\times(1-P)}$
文献[25]	4	$\frac{26496\times(1-P)+17056\times P}{4\times(1-P)}$
本文协议	5	$\frac{17120\times(1-P)+9216\times P}{5\times(1-P)}$

由此，在表 7 中列出了各认证协议在未知攻击下的通信成本。通信成本随 P 值的变化趋势情况如图 6 所示。

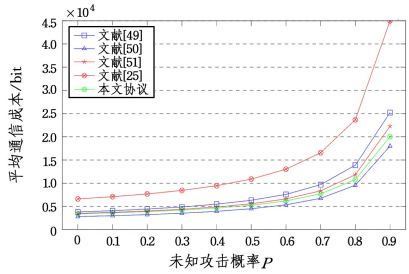


图 6 未知攻击下通信开销比较

Fig. 6 Comparison of communication overhead under unknown attack

结束语 本文提出了一种基于区块链技术的分布式云雾架构的智慧医疗安全认证密钥协商协议，协议可以抵抗云雾泄露攻击。协议中由多个云服务器共同维护智慧医疗区块链的账本，一方面避免了集中式架构容易导致的单点失效问题，另一方面能够抵抗各种已知攻击，即使在不完全可信的云雾节点数据泄露的情况下，也能保障隐私数据的安全。在安全分析过程中，使用扩展的 ROR 模型进行形式化安全证明，结果表明，提出的协议在该模型下具有会话钥语义安全性。启发式安全分析方法(即基于评价标准的安全分析)和 AVISPA 工具的仿真结果显示，协议能够抵抗各种已知攻击。性能比较分析表明，对比相关协议，在拥有较低的通信成本、计算成本、能源消耗和未知攻击下通信开销的情况下，该协议还能实现更多的功能与安全属性。尽管本文所提出的认证协议适用于云雾架构下的物联网应用场景，但在特定场景下，可能仍需依据实际情况进行调整与优化。特别是量子计算机的发展会对基于传统密码学的认证协议和应用产生巨大冲击，因此未来将聚焦于针对量子计算机的攻击者能力抵抗，研发更高效且安全的认证协议，以充分契合这些特定场景下的安全需求。

参 考 文 献

[1] WANG W M, HUANG H P, XIAO F, et al. Computation-trans-

ferable Authenticated Key Agreement Protocol for Smart Healthcare[J]. Journal of Systems Architecture, 2021, 118: 102215.

[2] XU Z S, XU J B, LI D K. A Token-based Authentication and Key Agreement Protocol for Cloud Computing[C]// 2021 IEEE 6th International Conference on Smart Cloud(SmartCloud). Piscataway, NJ: IEEE, 2021: 38-43.

[3] MOOKHERJI S, ODELU V, PRASATH R, et al. Fog-based Single Sign-on Authentication Protocol for Electronic Healthcare Applications[J]. IEEE Internet of Things Journal, 2023, 10(12): 10983-10996.

[4] HAYYOLALAM V, ALOQAILY M, ÖZKASAP Ö, et al. Edge-assisted Solutions for IoT-based Connected Healthcare Systems: A Literature Review[J]. IEEE Internet of Things Journal, 2021, 9(12): 9419-9443.

[5] KE C B, ZHU Z J, XIAO F, et al. SDN-based Privacy and Functional Authentication Scheme for Fog Nodes of Smart Healthcare[J]. IEEE Internet of Things Journal, 2022, 9(18): 17989-18001.

[6] GUO Y M, ZHANG Z F, GUO Y J. Secfhome: Secure Remote Authentication in Fog-enabled Smart Home Environment[J]. Computer Networks, 2022, 207: 108818.

[7] GUO Y M, GUO Y J. FogHA: An Efficient Handover Authentication for Mobile Devices in Fog Computing[J]. Computers & Security, 2021, 108: 102358.

[8] BONOMI F, MILITO R, ZHU J, et al. Fog Computing and Its Role in the Internet of Things[C]// Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing. New York: ACM, 2012: 13-16.

[9] SHI W S, CAO J, ZHANG Q, et al. Edge Computing: Vision and Challenges[J]. IEEE Internet of Things Journal, 2016, 3(5): 637-646.

[10] GUO Y M, ZHANG Z F, GUO Y J, et al. BSRA: Blockchain-based Secure Remote Authentication Scheme for the Fog-enabled Internet of Things[J]. IEEE Internet of Things Journal, 2024, 11(2): 3348-3361.

[11] GUO Y M, ZHANG Z F, GUO Y J. Fog-centric Authenticated Key Agreement Scheme without Trusted Parties[J]. IEEE Systems Journal, 2020, 15(4): 5057-5066.

[12] AMANLOU S, HASAN M K, BAKAR K A A. Lightweight and Secure Authentication Scheme for IoT Network Based on Publish-subscribe Fog Computing Model[J]. Computer Networks, 2021, 199: 108465.

[13] XU Z S, LIANG W, LI K, et al. A Blockchain-based Roadside Unit-assisted Authentication and Key Agreement Protocol for Internet of Vehicles[J]. Journal of Parallel and Distributed Computing, 2021, 149: 29-39.

[14] LI X C, YIN X C. Blockchain-based Group Key Agreement Protocol for Vehicular Ad Hoc Networks[J]. Computer Communications, 2022, 183: 107-120.

[15] CHATTARAJ D, BERA B, DAS A K, et al. Block-clap: Blockchain-assisted Certificateless Key Agreement Protocol for Internet of Vehicles in Smart Transportation[J]. IEEE Transactions on Vehicular Technology, 2021, 70(8): 8092-8107.

[16] LI J Y, QIAO Z Q, PENG J L. Asymmetric Group Key Agreement Protocol Based on Blockchain and Attribute for Industrial Internet of Things[J]. IEEE Transactions on Industrial Informatics, 2022, 18(11): 8326-8335.

[17] ARMANDO A, BASIN D, BOICHUT Y, et al. The AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications[C]// Computer Aided Verification: 17th International Conference. Berlin: Springer, 2005: 281-285.

[18] IBRAHIM M H. Octopus: An Edge-fog Mutual Authentication Scheme[J]. International Journal of Network Security, 2016, 18(6): 1089-1101.

[19] SRINIVAS J, DAS A K, KUMAR N, et al. Cloud Centric Authentication for Wearable Healthcare Monitoring System[J]. IEEE Transactions on Dependable and Secure Computing, 2018, 17(5): 942-956.

[20] WAZID M, DAS A K, KUMAR N, et al. Design of Secure Key Management and User Authentication Scheme for Fog Computing Services[J]. Future Generation Computer Systems, 2019, 91: 475-492.

[21] GUO Y M, ZHANG Z F, GUO Y J. Anonymous Authenticated Key Agreement and Group Proof Protocol for Wearable Computing[J]. IEEE Transactions on Mobile Computing, 2021, 21(8): 2718-2731.

[22] GUO Y M, GUO Y J. CS-LAKA: A Lightweight Authenticated Key Agreement Protocol with Critical Security Properties for IoT Environments[J]. IEEE Transactions on Services Computing, 2023, 16(6): 4102-4114.

[23] JIA X Y, HE D B, KUMAR N, et al. Authenticated Key Agreement Scheme for Fog-driven IoT Healthcare System[J]. Wireless Networks, 2019, 25(8): 4737-4750.

[24] MA M M, HE D B, WANG H Q, et al. An Efficient and Provably Secure Authenticated Key Agreement Protocol for Fog-based Vehicular Ad-hoc Networks[J]. IEEE Internet of Things Journal, 2019, 6(5): 8065-8075.

[25] LI X H, CHEN T, CHENG Q F, et al. An Efficient and Authenticated Key Establishment Scheme Based on Fog Computing for Healthcare System[J]. Frontiers of Computer Science, 2022, 16: 1-12.

[26] SHEN J, YANG H J, WANG A X, et al. Lightweight Authentication and Matrix-based Key Agreement Scheme for Healthcare in Fog Computing[J]. Peer-to-Peer Networking and Applications, 2019, 12: 924-933.

[27] KALARIA R, KAYES A S M, RAHAYU W, et al. A Secure Mutual Authentication Approach to Fog Computing Environment[J]. Computers & Security, 2021, 111: 102483.

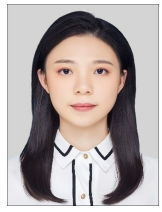
[28] YAO H L, YAN Q. Cryptographic Analysis and Design of Anonymous Authentication Protocol for Internet of Vehicles Va-lue added Service[J]. Journal of Computer Research and Deve-lopment, 2022, 59(2): 12.

[29] MA Y, SHI W, LI X, et al. Provable Secure Authentication Key Agreement for Wireless Body Area Networks[J]. Frontiers of Computer Science, 2024, 18(5): 185811.

[30] WANG Y, LIU Y. RC2PAS: Revocable Certificateless Conditional Privacy-preserving Authentication Scheme in WBANs[J]. IEEE Systems Journal, 2022, 16(4): 5675-5685.

[31] XIE X W, WU B, HOU B T. BEPHAP: A Blockchain-based Ef-

ficient Privacy-preserving Handover Authentication Protocol with Key Agreement for Internet of Vehicles[J]. Journal of Systems Architecture, 2023, 138: 102869.
[32] YU S, LEE J, SUTRALA A K, et al. LAKA-UAV: Lightweight Authentication and Key Agreement Scheme for Cloud-assisted Unmanned Aerial Vehicle Using Blockchain in Flying Ad-hoc Networks[J]. Computer Networks, 2023, 224: 109612.
[33] DONG J, XU G, MA C, et al. Blockchain-Based Certificate-Free Cross-Domain Authentication Mechanism for Industrial Internet [J]. IEEE Internet of Things Journal, 2024, 11(2): 3316-3330.
[34] WEI S J, LI S S, WANG J H. A Cross-domain Authentication Protocol by Identity-based Cryptography on Consortium Blockchain[J]. Chinese Journal of Computers, 2021, 44(5): 908-920.
[35] SHAO X W, GUO Y J. A Blockchain-based Authentication Protocol for Telemedicine[J]. Journal of Cryptologic Researchs, 2023, 10(2): 397-414.
[36] ZHENG Z B, XIE S A, DAI H N, et al. Blockchain Challenges and Opportunities: A Survey[J]. International Journal of Web and Grid Services, 2018, 14(4): 352-375.
[37] DOLEV D, YAO A. On the Security of Public Key Protocols [J]. IEEE Transactions on Information Theory, 1983, 29(2): 198-208.
[38] CANETTI R, KRAWCZYK H. Universally Composable Notions of Key Exchange and Secure Channels[C]// Advances in Cryptology—EUROCRYPT 2002: International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2002: 337-351.
[39] MESSERGES T S, DABBISH E A, SLOAN R H. Examining Smart-card Security Under the Threat of Power Analysis Attacks[J]. IEEE Transactions on Computers, 2002, 51(5): 541-552.
[40] WANG D, WANG P. Two Birds with One Stone: Two-factor Authentication with Security Beyond Conventional Bound[J]. IEEE Transactions on Dependable and Secure Computing, 2016, 15(4): 708-722.
[41] SHIHAB S, ALTAWY R. Lightweight Authentication Scheme for Healthcare with Robustness to Desynchronization Attacks [J]. IEEE Internet of Things Journal, 2023, 10(20): 18140-18153.
[42] WANG Q X, WANG D, CHENG C, et al. Quantum2FA: Efficient Quantum-resistant Two-factor Authentication Scheme for Mobile Devices[J]. IEEE Transactions on Dependable and Secure Computing, 2021, 20(1): 193-208.
[43] YANG H, GUO Y J, GUO Y M. Blockchain-based Cloud-fog Collaborative Smart Home Authentication Scheme[J]. Computer Networks, 2024, 242: 110240.
[44] WANG D, HE D B, WANG P, et al. Anonymous Two-factor Authentication in Distributed Systems: Certain Goals are Beyond Attainment[J]. IEEE Transactions on Dependable and Secure Computing, 2014, 12(4): 428-442.
[45] PARK K S, LEE J Y, DAS A K, et al. BPPS: Blockchain-enabled privacy-preserving scheme for demand-response management in smart grid environments[J]. IEEE Transactions on Dependable and Secure Computing, 2022, 20(2): 1719-1729.
[46] ZHANG S W, YAN Z W, LIANG W, et al. BAKA: Biometric authentication and key agreement scheme based on fuzzy extractor for wireless body area networks[J]. IEEE Internet of Things Journal, 2024, 11(3): 5118-5128.
[47] ARMANDO A, BASIN D, BOICHUT Y, et al. The AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications[C]// Computer Aided Verification: 17th International Conference. Berlin: Springer, 2005: 281-285.
[48] CHEVALIER Y, COMPAGNA L, CUELLAR J, et al. A High Level Protocol Specification Language for Industrial Security-sensitive Protocols[C]// Workshop on Specification and Automated Processing of Security Requirements (SAPS'2004). Austrian Computer Society, 2004: 13.
[49] FAN Q, CHEN J H, DEBORAH L J, et al. A Secure and Efficient Authentication and Data Sharing Scheme for Internet of Things Based on Blockchain[J]. Journal of Systems Architecture, 2021, 117: 102112.
[50] HEWA T, BRAEKEN A, LIYANAGE M, et al. Fog computing and blockchain-based security service architecture for 5G industrial IoT-enabled cloud manufacturing[J]. IEEE Transactions on Industrial Informatics, 2022, 18(10): 7174-7185.
[51] HUANG Y T, CHEN T S, WANG S D. Authenticated Key Agreement Scheme for Fog Computing in A Health-care Environment[J]. IEEE Access, 2023, 11: 46871-46881.
[52] YADAV A K, MISRA M, PANDEY P K, et al. An EAP-based Mutual Authentication Protocol for WLAN-connected IoT Devices[J]. IEEE Transactions on Industrial Informatics, 2022, 19(2): 1343-1355.
[53] XU Z S, LI X, XU J B, et al. A Secure and Computationally Efficient Authentication and Key Agreement Scheme for Internet of Vehicles[J]. Computers and Electrical Engineering, 2021, 95: 107409.
[54] CAO J, MA M D, FU Y L, et al. CPPHA: Capability-based Privacy-protection Handover Authentication Mechanism for SDN-based 5G HetNets[J]. IEEE Transactions on Dependable and Secure Computing, 2019, 18(3): 1182-1195.
[55] MA R H, CAO J, FENG D G, et al. FTGPHA: Fixed-trajectory Group Pre-handover Authentication Mechanism for Mobile Relays in 5G High-speed Rail Networks[J]. IEEE Transactions on Vehicular Technology, 2019, 69(2): 2126-2140.



YANG Xin, born in 2001, postgraduate, is a student member of CCF (No. Z1262G). Her main research interests include identity authentication and key agreement.

GUO Yimin, born in 1992. Ph. D, associate professor, master's supervisor, is a member of CCF (No. K7779S). Her main research interests include passwords, authentication protocol and modern cryptography.

(责任编辑:何杨)